

Classical and quantum algorithms for scaling problems

Harold Nieuwboer

Classical and quantum
algorithms for scaling problems

Harold Nieuwboer

Classical and quantum algorithms for scaling problems

Harold Adriaan Nieuwboer



Copyright © by Harold Nieuwboer.
Cover co-designed with Garazi Muguruza Lasa.

The author acknowledges support by the NWO through grant OCENW.KLEIN.267.

Classical and quantum algorithms for scaling problems

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. P.P.C.C. Verbeek
ten overstaan van een door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op woensdag 31 januari 2024, te 16.00 uur

door

Harold Adriaan Nieuwboer

geboren te Haarlem

Promotiecommissie

<i>Promotores:</i>	prof. dr. M. Walter prof. dr. E.M. Opdam	Ruhr-Universität Bochum Universiteit van Amsterdam
<i>Overige leden:</i>	prof. dr. H.M. Buhrman prof. dr. P. Bürgisser dr. D.N. Dadush dr. O. Fawzi prof. dr. C. Schaffner prof. dr. R.M. de Wolf dr. J. Zuiddam	Universiteit van Amsterdam Technische Universität Berlin CWI ENS Lyon Universiteit van Amsterdam Universiteit van Amsterdam Universiteit van Amsterdam

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

List of publications

This dissertation is based on the following papers. The authors of these papers have equally contributed to the obtained results.

- [BLNW20] **Interior-point methods for unconstrained geometric programming and scaling problems**
Peter Bürgisser, Yinan Li, Harold Nieuwboer, Michael Walter
arXiv:2008.12110, 2020
- [AGL+21] **Quantum Algorithms for Matrix Scaling and Matrix Balancing**
Joran van Apeldoorn, Sander Gribling, Yinan Li, Harold Nieuwboer, Michael Walter, Ronald de Wolf
48th International Colloquium on Automata, Languages, and Programming (ICALP), 2021, 110:1–110:17
arXiv:2011.12823, 2020
- [GN22] **Improved Quantum Lower and Upper Bounds for Matrix Scaling**
Sander Gribling, Harold Nieuwboer
39th International Symposium on Theoretical Aspects of Computer Science (STACS), 2022, 35:1–35:23
arXiv:2109.15282, 2021
- [AMN+23] **The minimal canonical form of a tensor network**
Arturo Acuaviva, Visu Makam, Harold Nieuwboer, David Pérez-Garcia, Friedrich Sittner, Michael Walter, Freek Witteveen
IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS), 2023, to appear
arXiv:2209.14358, 2022
- [AGN23] **Basic quantum subroutines: finding multiple marked elements and summing numbers**
Joran van Apeldoorn, Sander Gribling, Harold Nieuwboer
arXiv:2302.10244, 2023
- [HNW23] **Interior-point methods on manifolds: theory and applications**
Hiroshi Hirai, Harold Nieuwboer, Michael Walter
IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS), 2023, to appear
arXiv:2304.04771, 2023
This paper is the result of merging the two independent publications [Hir22b] and [NW23]. Results that were exclusively contributed by [Hir22b] are explicitly attributed as such.

Contents

1. Introduction	1
1.1. Examples of scaling problems and applications	3
1.2. Scaling, polytopes and invariants	6
1.3. Algorithms and obstructions to efficiency	10
1.4. Summary of results	12
I. Scaling problems and applications	19
2. Setting the stage	21
2.1. Notation and conventions	21
2.2. Preliminaries on algebra, geometry and groups	21
2.3. Geometry, orbits, and invariants	30
2.4. The Kempf–Ness theorem	41
2.5. The moment map	44
2.6. The computational problems	47
3. The minimal canonical form of a tensor network	53
3.1. Introduction	53
3.2. Matrix product states	60
3.3. Projected entangled pair states	72
3.4. Algorithms for computing minimal canonical forms	91
3.5. Conclusion and outlook	102
II. Interior-point methods for scaling	107
4. An introduction to interior-point methods	109
4.1. The idea	109
4.2. Self-concordant barriers	111
4.3. Following the central path	114
5. Interior-point methods for commutative scaling problems	119
5.1. Introduction	119
5.2. Summary of results	124
5.3. Condition measures and diameter bounds	128
5.4. Interior-point methods for unconstrained geometric programming	132
5.5. Bounds on condition measures	141
6. Preliminaries in Riemannian geometry	147
6.1. Metric, lengths, distances	147
6.2. Covariant derivative and curvature	148

6.3.	Parallel transport, geodesics, completeness	149
6.4.	Gradient and Hessian	151
6.5.	Convexity	151
7.	Interior-point methods on manifolds: overview	157
7.1.	Self-concordance and Newton’s method on manifolds	159
7.2.	Barriers and a path-following method on manifolds	160
7.3.	Examples of self-concordance: Squared distance in non-positive curvature	161
7.4.	Application I: Non-commutative optimization and scaling problems	164
7.5.	Application II: Minimum-enclosing ball problem on PD(n)	165
7.6.	Application III: Geometric median on hyperbolic space	166
7.7.	Outlook	167
8.	Interior-point methods on manifolds: the framework	169
8.1.	Self-concordance and Newton’s method	169
8.2.	Barriers, compatibility, and the path-following method	182
9.	Self-concordance of the squared distance in non-positive curvature	199
9.1.	Hadamard manifolds	199
9.2.	Positive definite matrices	201
9.3.	Constant negative curvature	212
10.	Interior-point methods for non-commutative scaling and geometric problems	223
10.1.	Non-commutative optimization and scaling problems	223
10.2.	The minimum enclosing ball problem	229
10.3.	The geometric median on model spaces	232
10.4.	The Riemannian barycenter	235
III.	Quantum algorithms and lower bounds for scaling	237
11.	An introduction to quantum algorithms and lower bounds	239
11.1.	Quantum computing	239
11.2.	Common subroutines	241
11.3.	Lower bound techniques	245
12.	Basic quantum subroutines, improved	247
12.1.	Introduction	247
12.2.	Preliminaries	251
12.3.	Fast Grover search for multiple items, without quantum memory .	252
12.4.	Improved query complexity for approximate summation	261
13.	Matrix scaling and matrix balancing	267
13.1.	Introduction	267
13.2.	Preliminaries	277
13.3.	Quantum subroutines for matrix scaling and balancing	283

14. Quantum Sinkhorn and Osborne algorithms	297
14.1. Quantum Sinkhorn algorithm	297
14.2. Improved analysis for entrywise-positive matrices	306
14.3. Randomized quantum Sinkhorn algorithm	312
14.4. Randomized quantum Osborne algorithm	317
15. Quantum box-constrained Newton methods	327
15.1. Minimizing second-order robust convex functions	327
15.2. Quantum box-constrained matrix scaling	332
15.3. Quantum box-constrained matrix balancing	341
16. Quantum query lower bounds: constant precision	349
16.1. Partially learning a string hidden in a permutation	349
16.2. Lower bound for matrix scaling	352
16.3. Lower bound for matrix balancing	354
17. Quantum query lower bounds: high precision	359
17.1. The basic lower bound	359
17.2. Definition of the scaling instances and analysis of row marginals . .	360
17.3. Concentration of column marginals	361
17.4. Strong convexity properties of the potential	363
17.5. Concluding the lower bound for matrix scaling	366
17.6. Lower bound for matrix balancing	367
17.7. Lower bound for computing the row marginals	373
Bibliography	375
Abstract	395
Samenvatting	397
Acknowledgements	399

1. Introduction

This thesis is concerned with *scaling problems*, which have been of much interest in recent years. It is a class of computational problems with a plethora of connections to different areas of mathematics, physics and computer science. Although many structural aspects of these problems are understood by now, we only know how to solve them *efficiently* in special cases.

To demonstrate the breadth of this subject, we mention some applications: approximating the permanent [LSW00], non-commutative rational identity testing [GGOW16], Brascamp–Lieb inequalities [GGOW18], Horn’s problem on spectra of sums of Hermitian matrices [Fra18], the Paulsen problem [KLLR18; HM21b], strengthening the Sylvester–Gallai theorem [BDWY12; DSW14; DGOS18], lower bounds on unbounded-error communication complexity [For01], approximating optimal transport plans in machine learning [Cut13], maximum-likelihood estimation in statistics [AKRS21b; AKRS21a; DM21; DMW22; FORW21], the quantum marginal problem [Kly02; Kly04; Kly06; BGO+18; BFG+18], asymptotic non-vanishing of Kronecker coefficients in representation theory [IMW17; BFG+18], and geometric invariant theory [KN79; NM84; MFK94]. We refer to [BFG+19] for a more complete overview and the history of scaling problems. Related *orbit problems* also have strong connections to Mulmuley and Sohoni’s geometric complexity theory approach to Valiant’s VP versus VNP [Val79; MS08; BLMW11; Mul17; IP17; BIP19; DIP20], and various notions of tensor rank and the complexity of matrix multiplication [Str86; Str87; Str88; Str91; Lan17; CVZ21; BIL+21; Der22].

The primordial example of a scaling problem is that of *matrix scaling*, after which this class of problems is named (with the problem itself going back to Kruihof [Kru37] in 1937, and the terminology dating back to at least 1968 [MO68]). Its statement is deceptively simple: given a matrix with non-negative real entries, rescale its rows and columns by positive numbers, such that the resulting matrix has all row and column sums 1, i.e., the rescaled matrix is *doubly stochastic*. A *non-commutative* version of this problem called *operator scaling* was introduced by Gurvits [Gur04] in the context of Edmonds’ problem. Here, one is asked to “rescale” a completely positive map such that it becomes unital and trace-preserving; this can be viewed as a “quantum generalization” of double stochasticity. This can be further generalized to the *tensor scaling* problem, where one has to convert a pure multipartite quantum state to a quantum state whose one-body marginals are proportional to the identity matrix, using only a restricted set of operations. Such a generalization arises naturally in the context of understanding the entanglement of quantum states [Kly02; Kly04; BGO+18; BFG+18]. We discuss matrix and tensor scaling in more detail in Section 1.1.

As elucidated in a long sequence of works and explained in Section 1.2, these problems and many others can be solved by solving a *norm minimization* problem: given a linear action of a “nice” group on a suitably normed vector space, and

This chapter is partially adapted from [BLNW20; AGL+21; AMN+23; HNW23].

1. Introduction

a vector therein, the goal is to find a vector of minimal norm in its orbit (or the closure thereof). The celebrated Kempf–Ness theorem [KN79] states that such a minimum norm vector is exactly the solution to the scaling problem! This is an important result in the area of geometric invariant theory, which can also be used to understand the structure of scaling problems. For instance, whether a scaling problem admits a solution at all is governed by certain *invariant polynomials* [MFK94; KN79]. This connection is useful in the context of analyzing the performance of algorithms for scaling.

When the group is commutative, such as in the case of matrix scaling, norm minimization problems reduce to (*unconstrained*) *geometric programs*, a well-known generalization of linear programming. After a suitable change of coordinates, such programs are convex, and can be solved efficiently using techniques from convex optimization [NR99; SV14; CMTV17; ALLOW17; BLNW20].

In operator scaling and tensor scaling, which capture many of the previously mentioned problems, the group is non-commutative; hence this class of problems has also been called *non-commutative (group) optimization* problems. There is again a relevant notion of convexity: the norm minimization problem is a *geodesically convex* optimization problem on a *homogeneous space of non-positive curvature*. Currently, the best algorithms exploit this geodesic convexity and geodesic generalizations of convex programming techniques to give algorithms with provable guarantees. However, efficient algorithms are known only in special cases, which have recently been understood to satisfy a certain total unimodularity property [BFG+19].

This thesis contributes to this area in various ways:

Interior-point methods for scaling. We give new algorithms for non-commutative scaling problems with complexity guarantees that match the prior state of the art. To this end, we extend the well-known (self-concordance based) *interior-point method* (IPM) framework to the setting of Riemannian manifolds. This approach is particularly motivated by the fact that – as we also show – IPMs give efficient algorithms for commutative scaling problems. Moreover, the IPM framework does not obviously suffer from the same obstructions as previous methods, which we discuss in more detail in Section 1.3. It also yields the first high-precision algorithms for other natural geometric problems such as computing geometric medians and minimum-enclosing balls on symmetric spaces of non-positive curvature.

Quantum algorithms for scaling. For the important (commutative) problems of matrix scaling and balancing, we show that one can leverage the power of quantum computation to outperform the (already very efficient) state-of-the-art classical algorithms. In certain parameter regimes, this yields algorithms which can solve the matrix scaling problem in time *sublinear* in the size of the input matrix, when one is given quantum query access to the matrix; classically, this is impossible, as one has to at least read the input to the problem! We also show that in certain regimes our quantum algorithms are optimal, and in other regimes no quantum speedup over the classical methods is possible. Along the way, we provide improvements over the long-standing state of the art for basic quantum subroutines, such as searching for all marked elements in a list, and computing the sum of a list of numbers.

Scaling for tensor networks. We also identify a new application in the context of tensor networks for quantum many-body physics. We use the theory to define a canonical form for uniform projected entangled pair states, circumventing previously known undecidability results. Computing the canonical form amounts to solving a norm minimization problem, or equivalently a scaling problem, and we give algorithms with rigorous complexity guarantees for doing so. We also show, by characterizing the invariant polynomials, that the canonical form is determined by evaluating the tensor network contractions on networks of bounded size.

Organization. The rest of this introduction is organized as follows. In Section 1.1 we discuss matrix and tensor scaling in more detail. In Section 1.2 we informally define the general scaling problem, give an overview of its structural properties, and hint at the connection to geometric invariant theory. Next, in Section 1.3 we discuss the current state-of-the-art for algorithms for scaling problems, and obstructions to providing efficient algorithms for the general setting. Finally, in Section 1.4 we give a more precise overview of the contributions in this thesis.

1.1. Examples of scaling problems and applications

Matrix scaling. Let $A \in \mathbb{R}_{\geq 0}^{n \times n}$ be a matrix with non-negative entries. Then the *matrix scaling* problem is to rescale the rows and columns of A so that its row and column sums are approximately given by $\mathbf{1}$ and $\mathbf{1}$, respectively. That is, we wish to find positive diagonal matrices $X, Y \in \mathbb{R}^{n \times n}$ such that XAY is approximately *doubly stochastic*. Observe that the set of pairs (X, Y) with X, Y positive diagonal matrices forms a group under matrix multiplication. This group acts on $A \in \mathbb{R}_{\geq 0}^{n \times n}$ by left- and right-multiplying, and we must find a matrix with certain row and column sums in the orbit A ; hence we have a group, a representation and a vector in it.

This problem is very well-studied and has a wide range of applications. It was introduced by Kruithof for Dutch telephone traffic computation [Kru37], and has also been used in other areas of economics [Sto64]. In mathematics, it has been used as a common tool in practical linear algebra computations [LG04; Bra10; PC11; OCPB16], but also in statistics [Sin64], optimization [RS89], optimal transport [Cut13], and for strengthening the Sylvester-Gallai theorem [BDWY11]. Matrix scaling can be solved in polynomial time [KK96; NR99], and deciding scalability can even be done in strongly polynomial time [LSW00]. More recent works even provide near-linear time algorithms under reasonable assumptions [ALOW17; CMTV17; CKL+22; BCK+23]. We refer to [Ide16] for a survey of matrix scaling, its applications and some history.

A common approach to solving matrix scaling problems is Sinkhorn’s algorithm, which is a simple iterative procedure, which alternates between scaling the rows sums and the column sums to the desired marginals:

- (i) Initialize $X, Y = I_n$ to the identity matrix.
- (ii) Update X_i by $X_i \leftarrow 1 / (\sum_{j=1}^n X_i A_{ij} Y_j)$ for $i \in [n]$.
- (iii) Update Y_j by $Y_j \leftarrow 1 / (\sum_{i=1}^n X_i A_{ij} Y_j)$ for $j \in [n]$.
- (iv) Go back to step (ii).

1. Introduction

The update rule in (ii) is such that at the end of this step, XAY has row sums equal to 1, and similarly, XAY has column sums equal to 1 at the end of (iii). Surprisingly, this algorithm converges to an actual solution whenever one exists. This can be (morally) justified by the following argument: let $\mathbf{1} \in \mathbb{R}^n$ be the all-ones vector, and consider the function $f: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(x_1, \dots, x_n, y_1, \dots, y_n) = \sum_{i,j=1}^n A_{ij} e^{x_i + y_j} - \langle x, \mathbf{1} \rangle - \langle y, \mathbf{1} \rangle. \quad (1.1.1)$$

Then f is a *convex function*, so its critical points are automatically minimizers. One can check that the gradient $\text{grad } f(x, y)$ of f at (x, y) is zero if and only if $\text{diag}(e^x)A \text{diag}(e^y)$ is doubly stochastic. In other words, f acts as a *convex potential function* for the matrix scaling problem. Furthermore, Sinkhorn's algorithm can be seen as performing *block coordinate descent* with respect to the variables x_1, \dots, x_n and y_1, \dots, y_n . The convexity of f then suggests that this algorithm should converge whenever an (approximate) scaling of A exists. This function can also be used to give *quantitative* bounds on the convergence speed. The state-of-the-art algorithms for solving the matrix scaling problem (as well as the very similar *matrix balancing* problem) are based on minimizing f or closely related functions [CMTV17; ALOW17; CKL+22].

It turns out that it is also easy to characterize *when* a matrix A is scalable [RS89], and this is governed by its *support*: A is (exactly) scalable if and only if the tuple $(\mathbf{1}/n, \mathbf{1}/n)$ is in (the relative interior of) the *convex hull* of the vectors $(e_i, e_j) \in \mathbb{R}^n \times \mathbb{R}^n$ such that $A_{ij} > 0$. In fact, this convex hull is exactly the set of achievable row and column sums (after normalizing). Therefore, determining whether A is scalable is a *convex polytope membership problem*, and may be solved using linear programming techniques. This condition also has a combinatorial interpretation: if A is viewed as a weighted bipartite adjacency matrix, then A is scalable if and only if the corresponding graph contains a bipartite perfect matching.

Tensor scaling and quantum marginals. Given density matrices ρ_1, \dots, ρ_k , each describing the quantum state of one party, does there exist a k -party pure quantum state with marginals equal to the ρ_k ? This problem is called the one-body quantum marginal problem, and is a special case of the *tensor scaling* problem, which is as follows. For simplicity, we take $k = 3$ parties of the same dimension $n \geq 2$. Let $|\psi\rangle \in V := \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$ be a pure state, and let $g = (g_1, g_2, g_3) \in G := \text{SL}(n, \mathbb{C})^3$. Consider the (unnormalized) pure state $|\psi_g\rangle = (g_1 \otimes g_2 \otimes g_3)|\psi\rangle$, let $\rho_g = |\psi_g\rangle\langle\psi_g| / \langle\psi_g|\psi_g\rangle$ be the associated density matrix, and let $\rho_{g,j} = \text{Tr}_{j^c}[\rho_g]$ denote its reduced density matrix on the j -th subsystem for $j \in [3]$ (obtained by taking the partial trace over the other two subsystems). Then what are the possible achievable triples of reduced density matrices $(\rho_{g,1}, \rho_{g,2}, \rho_{g,3})$, as g ranges over $\text{SL}(n, \mathbb{C})^3$?

This question has an operational interpretation: the states $|\psi_g\rangle$ are precisely those that can be made from $|\psi\rangle$ using *stochastic local operations and classical communication* (SLOCC). The “local operations” part here refers to the fact that we are only allowed to act with tensor products $g_1 \otimes g_2 \otimes g_3$. The “stochasticity and classical communication” part amounts to allowing local measurements with post-selection on joint outcomes, such that the overall protocol succeeds with

some non-zero probability. This naturally yields local operations in $GL(n, \mathbb{C})$; however, since we do not care about the overall normalization, we are free to work with $SL(n, \mathbb{C})$ instead.

We explain how to relate this problem to a *norm minimization* problem, for the special case of whether $(I/n, I/n, I/n)$ is achievable. Consider the following minimization of the (for convenience squared) ℓ^2 -norm over all rescalings of $|\psi\rangle$:

$$\inf_{g_1, g_2, g_3 \in SL(n, \mathbb{C})} \|(g_1 \otimes g_2 \otimes g_3) |\psi\rangle\|_2^2. \quad (1.1.2)$$

Then the (logarithmic) gradient of the objective at $g = (g_1, g_2, g_3)$ is given by $(\rho_{g,1} - I/n, \rho_{g,2} - I/n, \rho_{g,3} - I/n)$. As minimizers of Eq. (1.1.2) are critical points of the objective, i.e., points where the gradient vanishes, these correspond to quantum states with maximally mixed marginals. The quantum marginal problem amounts to characterizing the set of possible gradients for *generic* $|\psi\rangle$. It turns out that the set of *sorted eigenvalues* of the achievable gradients is a convex polytope [Kly04; BGO+18; BFG+18]! This is a special case of a much more general phenomenon [NM84; GS84; Kir84a; Bri88]. Therefore the existence of a pure state with given local spectra is governed by a *finite number of linear inequalities* on the eigenvalues. We explain this in more detail in Section 1.2. However, we remark here that these inequalities are not *computationally* useful, since it is difficult to enumerate them [Kly04; VW17], and hence determining whether a joint spectrum can arise requires other approaches.

We can reduce the domain of optimization to $M = SPD(n) \times SPD(n) \times SPD(n)$, where $SPD(n)$ denotes the complex positive-definite matrices of unit determinant: since $P_j := g_j^* g_j$ is an arbitrary matrix in $SPD(n)$, Eq. (1.1.2) is equivalent to:

$$\inf_{P_1, P_2, P_3 \in SPD(n)} \langle \psi | (P_1 \otimes P_2 \otimes P_3) | \psi \rangle \quad (1.1.3)$$

Unfortunately, the domain is non-convex as a subset of the Euclidean space of triples of Hermitian matrices, and in any case the objective would not be a convex function of the variables if relaxed to $PD(n)$, so it is not clear that one could use standard techniques such as semidefinite programming to solve this problem. Moreover, the naive exponential reparameterization $P_i = e^{H_i}$ with Hermitian H_i does not yield a convex problem in the H_i either, because the matrix exponential is not operator convex [Bha13, Prob. V.5.1].

However, a key observation is that the objective becomes convex when $SPD(n)$ and hence M is given a natural non-Euclidean geometry, namely the so-called *affine-invariant* metric, which also appears as the *Fisher-Rao metric* for Gaussian covariance matrices in statistics (see Section 7.3 for a precise definition). Then the straight lines of Euclidean space get replaced by the geodesics of the new metric, which take the form $P_j(t) = \sqrt{P_j} e^{tH_j} \sqrt{P_j}$ for traceless Hermitian matrices H_j and clearly remain in $SPD(n)$. It is easy to verify that the objective in Eq. (1.1.3) is *convex along such geodesics* (in fact, log-convex). This is the “correct” non-commutative generalization of the fact that $f(x, y)$ in Eq. (1.1.1) is convex, but would not be without the change of coordinates $e^{x_i} = X_i$, $e^{y_j} = Y_j$.

The tensor scaling problem is not currently known to be solvable in polynomial time in all parameters, although partial results are known [BFG+18], which we elaborate on later. There is complexity-theoretic evidence that polynomial-time algorithms might exist, as the one-body quantum marginal problem (i.e., tensor scaling for generic $|\psi\rangle$) is in $NP \cap coNP$ [BCMW17].

1.2. Scaling, polytopes and invariants

We now explain on a high level the connection between the previous examples, and define the general scaling and norm minimization problems. In each of the examples, there is some initial data: a matrix in the case of matrix scaling, and a multipartite quantum state in the case of tensor scaling. They are transformed according to a certain set of allowed operations, which in each case forms a group: pairs (X, Y) of positive diagonal matrices in the case of matrix scaling, and triples of invertible matrices in $SL(n, \mathbb{C})$ in the case of tensor scaling. The goal is to reach a certain *scaling* or “marginal condition”: having certain row and column sums, or the quantum state having maximally mixed reduced one-body density matrices.

These fit together in a general framework as follows [BFG+19]: we are given a group G , a representation $\pi: G \rightarrow GL(V)$ of G , and some vector $v \in V \setminus \{0\}$. We write $g \cdot v = \pi(g)v$ for the result of acting with g on a vector $v \in V$. The group G is assumed to be a *connected complex reductive algebraic group*, given to us as a subgroup $G \subseteq GL(n, \mathbb{C})$ satisfying explicit polynomial equations. Examples of such G are $GL(n, \mathbb{C})$ itself, the group $SL(n, \mathbb{C})$ consisting of determinant-one matrices, the special orthogonal group $SO(n, \mathbb{C})$, the symplectic group $Sp(2n, \mathbb{C})$, and products of these groups. Particularly important is the group of diagonal matrices $(\mathbb{C}^\times)^n = GL(1, \mathbb{C})^n$, and all relevant *commutative* G are in fact isomorphic to this group. In more detail, we assume that G is closed under adjoints ($g \in G$ implies $g^* = \bar{g}^\top \in G$), that the action π is *regular* (given by polynomials) and that V is a complex Hilbert space such that the subgroup $K = G \cap U(n)$ of G acts on V by unitary matrices.

The objective for the norm minimization problem is then the *Kempf–Ness* function $F_v: G \rightarrow \mathbb{R}$, defined by

$$F_v(g) = \log \|g \cdot v\|, \quad F_v^* = \inf_{g \in G} F_v(g). \quad (1.2.1)$$

Note that $g \cdot v$ is never the zero vector, so the above is well-defined. The infimum $\inf_{g \in G} \|g \cdot v\| = e^{F_v^*}$ is sometimes called the *capacity* $\text{cap}(v)$ of v [Gur04; BGO+18; BFG+19]. Then we define:

Problem 1.2.1 (Norm minimization). *Let $v \in V \setminus \{0\}$ and $\delta > 0$. Then the norm minimization problem for v is to find $g_\delta \in G$ such that $F_v(g_\delta) \leq F_v^* + \delta$, or to assert that F_v is unbounded from below.*

Clearly, if the Kempf–Ness function has a minimizer, then there must be some point where its *gradient* vanishes. This gradient lives in the Lie algebra $\text{Lie}(G) \subseteq \mathbb{C}^{n \times n}$ of G ; informally, this is the set of infinitesimal directions at $I \in G$. We write $\text{Lie}(K) \subseteq \text{Lie}(G)$ for the Lie algebra of K defined similarly, and $i\text{Lie}(K) = \{iX : X \in \text{Lie}(K)\} \subseteq \text{Lie}(G)$. In the case of $K = U(n)$, $\text{Lie}(K)$ is given by the skew-Hermitian matrices, so $i\text{Lie}(K)$ consists of the Hermitian matrices.

The scaling problem then arises naturally as the problem of minimizing the norm of the gradient, and formally defined as follows. Define the *moment map* $\mu: V \setminus \{0\} \rightarrow i\text{Lie}(K)$ by letting $\mu(v) \in \text{Lie}(G)$ be the unique matrix satisfying

$$\text{Tr}[\mu(v)H] = \partial_{t=0} \log \|e^{tH} \cdot v\|$$

for all $H \in \text{Lie}(G)$. In other words, $\mu(v)$ is the gradient of $F_v: G \rightarrow \mathbb{R}$ at the identity $I \in G$. Then $\mu(v)$ naturally takes values in $i\text{Lie}(K)$: K acts unitarily on V ,

hence preserves the norm, so the only way to change F_v is to move in directions orthogonal to $\text{Lie}(K)$, and its orthogonal complement is exactly $i\text{Lie}(K)$.

Problem 1.2.2 (Scaling). *Let $v \in V \setminus \{0\}$ and $\varepsilon > 0$. Then the scaling problem for v is to find $g_\varepsilon \in G$ such that $\|\mu(g_\varepsilon \cdot v)\|_{\text{HS}} \leq \varepsilon$, or to assert that no such g_ε exists.*

To see that the matrix scaling problem is indeed a special case of the above requires a (small) effort; in particular, there is some disconnect between the fact that we work over the complex numbers here, and over non-negative real numbers for matrix scaling. Let $G = \text{ST}(n) \times \text{ST}(n)$ be the group of pairs of diagonal $n \times n$ matrices with entries in \mathbb{C}^\times and determinant 1, and let G act on $V = \mathbb{C}^{n \times n}$ via

$$(X, Y) \cdot B = XBY = \begin{bmatrix} X_1 & & \\ & \ddots & \\ & & X_n \end{bmatrix} B \begin{bmatrix} Y_1 & & \\ & \ddots & \\ & & Y_n \end{bmatrix},$$

i.e., by rescaling the rows and columns by X and Y respectively. Endowing V with the Hilbert–Schmidt inner product, we see that

$$F_B(X, Y) = \log \|XBY\|_{\text{HS}} = \frac{1}{2} \log \sum_{i,j=1}^n |X_i B_{ij} Y_j|^2$$

and the moment map at B , i.e., the gradient of F_B at $(X, Y) = (I, I)$ is given by

$$\text{grad}_{X,Y=I} F_B(X, Y) = \frac{\sum_{i,j=1}^n |B_{ij}|^2 (\text{diag}(e_i), \text{diag}(e_j))}{\|B\|_{\text{HS}}^2} - (I/n, I/n).$$

The correction $-(I/n, I/n)$ appears because the X and Y are constrained to have determinant 1, hence infinitesimal changes in X and Y are restricted to matrices with trace 0. Observe now that the moment map $\mu((X, Y) \cdot B)$ is zero if and only if the matrix A with entries $A_{ij} = |X_i|^2 |B_{ij}|^2 |Y_j|^2$ has row and column sums $1/n$. Furthermore, whether this condition holds depends only on the absolute values of the entries of B , X , and Y , explaining why the matrix scaling problem only involves non-negative numbers.

We motivated the scaling problem above by asserting that if the norm minimization problem has an exact minimizer, then the gradient of the Kempf–Ness function must vanish somewhere, and so the scaling problem is solvable for $\varepsilon = 0$. In fact, the scaling problem can be solved for all $\varepsilon > 0$ if and only if the norm minimization problem is solvable for all $\delta > 0$, i.e., the Kempf–Ness function is bounded from below [KN79]:

Theorem 1.2.3 (Kempf–Ness). *The Kempf–Ness function F_v is bounded from below if and only if $0 \in \mu(G \cdot v)$. Furthermore, its minimum is attained if and only if $0 \in \mu(G \cdot v)$.*

The second part of the theorem is a consequence of the *geodesic convexity* of F_v . We briefly explain this. Since the action of $K = G \cap \text{U}(n)$ preserves the inner product and hence the norm on V , the Kempf–Ness function is also naturally defined on the quotient space $K \backslash G$ consisting of the right-cosets of K . This space is a simply connected *symmetric space*, and can be endowed with a natural Riemannian metric such that it has *non-positive curvature*. The geodesics (straight lines) with respect to

1. Introduction

this metric are curves of the form $t \mapsto Ke^{tH}g$, where $H \in \mathfrak{iLie}(K)$ and $g \in G$. Then the crucial point is that F_v is *convex* along these geodesics:

$$\partial_t^2 F_v(Ke^{tH}g) \geq 0.$$

Any two points on $K \backslash G$ are connected by a (unique) geodesic as well. Together with the convexity, this makes it easy to see that the critical points of F_v are automatically *global* minimizers.

To understand why the theorem holds beyond the case where F_v has an exact minimizer, i.e., where one only assumes that F_v is bounded from below, requires more input from *geometric invariant theory* (GIT). The following is a central definition in GIT:

Definition 1.2.4 (Stability and null cone). We say that $v \in V$ is *semistable* if $0 \notin \overline{G \cdot v}$, where $\overline{G \cdot v} \subseteq V$ is the closure of the G -orbit of v . Equivalently, F_v is bounded from below.

If v is not semistable, then v is called *unstable*. The set $\mathcal{N} = \{v \in V : v \text{ unstable}\}$ is called the *null-cone* of the representation.

The null-cone can be seen as the set of “bad vectors” in the context of forming *quotients of projective varieties*, as we shall explain in Section 2.3.4. With this definition and the Kempf–Ness theorem in hand, the decision variant of the norm-minimization and scaling problems is the following question: for a given $v \in V$, is v semistable? In the rest of this section we explain why one can hope to algorithmically solve this problem at all.

Moment polytopes. It turns out that characterizing when $0 \in \overline{\mu(G \cdot v)}$ has a rather combinatorial nature. When G is commutative, i.e., isomorphic to $(\mathbb{C}^\times)^n$, then the representation V is characterized by a finite set of integer vectors $\Omega \subset \mathbb{Z}^n$ called the *weights* of the representation. Every vector v can be decomposed as a sum $v = \sum_{\omega \in \Omega} v_\omega$ such that $z = (z_1, \dots, z_n) \in G$ acts on v_ω by multiplication with $z^\omega = z_1^{\omega_1} \cdots z_n^{\omega_n}$. The closure of the image of the moment map $\mu(G \cdot v)$ is then characterized as follows: it is the convex hull of those $\omega \in \Omega$ for which $v_\omega \neq 0$. In particular it is a *convex polytope*, usually referred to as the *moment polytope*. In the case of matrix scaling, this is the set of (asymptotically) achievable pairs (r, c) of row and column sums [RS89].

In the non-commutative setting, a similar result holds. Then the closure of the intersection $\Delta(v) = \overline{\mu(G \cdot v) \cap C^+}$ with a *positive Weyl chamber* C^+ is again a convex polytope [NM84; GS84; Bri87], called the moment polytope of v . For the tensor scaling problem, intersecting with the positive Weyl chamber amounts to computing the ordered spectrum of each of the one-body reduced density matrices, so the moment polytope consists of possible joint spectra (and their limits) achievable by SLOCC operations on a starting state $v = |\psi\rangle$ [Kly02; Kly04].

A quantitative Kempf–Ness theorem. An important property is now that, because there are only finitely many weights, there are only finitely many possible $\Delta(v)$ for a given representation V ! As a consequence, either $0 \in \Delta(v)$ and v is semistable, or the distance between 0 and $\Delta(v)$ is lower bounded by a constant that depends only

on the representation $\pi: G \rightarrow GL(V)$. A slight relaxation of this distance is known as the *weight margin* $\gamma(\pi) > 0$. Note that if $\|\mu(g \cdot v)\| < \gamma(\pi)$, then this $g \in G$ can be viewed as a *certificate* that v is semistable!

The weight margin plays an important role in the analysis of the algorithms for norm-minimization and scaling problems, and makes an appearance in *diameter bounds* on approximate minimizers. It also appears in a *quantitative* version of the Kempf–Ness theorem: vectors which are approximately scaled also have nearly minimal norm [BFG+19], with the conversion between these two errors depending on $\gamma(\pi)$.

One important situation in which $\gamma(\pi)$ is only inverse-polynomially small (as compared to inverse-exponentially) is when the matrix whose rows are given by the weights $\omega \in \Omega$ of the representation is *totally unimodular*. This combinatorial criterion guarantees that the *facets* of $\Delta(v)$ are not “too complicated”. As a consequence, either 0 is not in the moment polytope, or it is “far away” from it. This happens in various situations of interest, such as matrix scaling and balancing and operator scaling (and more generally for quiver representations), but notably not for tensor scaling.

There is also another direction for the quantitative Kempf–Ness theorem: vectors with close to minimal norm are approximately scaled. The parameter which appears in this conversion is the *weight norm* $N(\pi)$. This is the largest norm the image of the moment map can take, i.e., $\sup_{v \in V \setminus \{0\}} \|\mu(v)\|_{\text{HS}}$. Its name comes from the fact that this is also the largest norm of a weight of the representation V . One can show that the Kempf–Ness function is Lipschitz with $N(\pi)$ as Lipschitz constant, and the weight norm is also useful for bounding its higher-order derivatives (in particular the function is *smoothly* convex as we shall see later). Moreover, $N(\pi)$ is generally small for representations of interest (polynomially bounded in the input size, and sometimes even constant).

Scaling problems and invariant theory. Feasibility of the scaling problem is also intricately related to invariant theory, and this plays a key role in the runtime analysis of the algorithms. A classical result due to Mumford [MFK94] gives an equivalent criterion for v being in the null-cone. Let $\mathbb{C}[V]$ denote the *ring of polynomials* on V , and let $\mathbb{C}[V]^G$ denote the *invariant polynomials* on V , i.e., the set of those $p \in \mathbb{C}[V]$ such that $p(g \cdot w) = p(w)$ for all $w \in V$ and $g \in G$. It is clear that if v is in the null-cone, then $p(0) = p(v)$ for all $p \in \mathbb{C}[V]^G$; after all, polynomials are continuous, and $0 \in \overline{G \cdot v}$. More generally, if $v, w \in V$ are two vectors, then one can ask whether $\overline{G \cdot v}$ and $\overline{G \cdot w}$ have a non-empty intersection; if this is the case, then $p(v) = p(w)$ for all $p \in \mathbb{C}[V]^G$. Remarkably, this is also a sufficient condition:

Theorem 1.2.5 (Mumford). *Let $v, w \in V$. Then $\overline{G \cdot v} \cap \overline{G \cdot w} \neq \emptyset$ if and only if $p(v) = p(w)$ for all $p \in \mathbb{C}[V]^G$.*

One of the important properties of the ring of invariants is that it is also *finitely generated* as an algebra:

Theorem 1.2.6 (Hilbert). *There exist finitely many $p_1, \dots, p_r \in \mathbb{C}[V]^G$ such that every $p \in \mathbb{C}[V]^G$ is a polynomial in the p_j .*

This suggests that determining whether v is in the null-cone is a decidable problem. Indeed, one can compute generators p_1, \dots, p_r as above [DK15], then test

1. Introduction

whether $p_j(v) = p_j(0)$ for all $j = 1, \dots, r$. However, this approach is computationally infeasible: there can be too many generators, their degree may be high, or there may be complexity-theoretic obstructions to evaluating them efficiently [GIM+20]. We discuss this in more detail in Section 2.6. Nevertheless the above criterion is useful for *analyzing* algorithms for the norm-minimization and scaling problems. One way in which the above enters is that if one can show that there exists a generating set of polynomials p_1, \dots, p_r with bounded integral coefficients (in some basis) and a suitable bound on their *degrees*, then this yields a priori estimates on the values of F_v^* ; see [BFG+19, Sec. 7] for the general approach and Proposition 3.4.2 for an example.

1.3. Algorithms and obstructions to efficiency

We now turn to algorithms for solving scaling problems, highlighting the different approaches that have been taken so far in the literature. We also discuss certain *geometric* obstructions to giving efficient algorithms for general scaling problems.

Alternating minimization. In the case of matrix scaling, we saw that the simple iterative Sinkhorn’s algorithm is capable of finding solutions. This holds more generally for the *operator scaling* and tensor scaling problems (discussed above). The fundamental structure that is used here is that the domain is a product space, and that optimizing over one factor is easy. Such algorithms are more generally known as *alternating minimization* or *coordinate descent* algorithms, and can be analyzed for many examples, such as matrix scaling, operator scaling and tensor scaling [LSW00; Gur04; GGOW20; BGO+18; BFG+18]. Unfortunately, not every scaling problem admits such a structure.

Gradient descent. As the general optimization problem is (geodesically) convex, one may hope to apply standard convex optimization techniques. One such standard technique is gradient descent. It turns out that one can analyze gradient descent algorithms in our setting, and this yields efficient algorithms in some parameters [BFG+19, Thm. 4.2]:

Theorem 1.3.1 (Gradient descent). *Given semistable $v \in V \setminus \{0\}$ and $\varepsilon > 0$, there exists an algorithm that solves the scaling problem in*

$$O\left(\frac{N(\pi)^2}{\varepsilon^2}(F_v(I) - F_v^*)\right)$$

iterations. Every iteration consists of computing the gradient of the Kempf–Ness function and basic linear algebraic operations.

This follows rather straightforwardly from F_v being *smoothly* geodesically convex with smoothness parameter $O(N(\pi)^2)$; this quantity is an upper bound on the second derivative of F_v along a unit speed geodesic. The parameter $N(\pi)$ is usually small (polynomial in the input size) and the same holds for the *potential gap* $F_v(I) - F_v^*$. However, the dependence on the achieved precision $\varepsilon > 0$ is typically unsatisfactory: it is polynomial in $1/\varepsilon$ rather than $\log(1/\varepsilon)$ (even in the Euclidean

setting). This is not enough for certain applications, such as deciding whether 0 is in the moment polytope $\Delta(v)$ in polynomial time: after all, here one needs to pick ε of size roughly the weight margin $\gamma(\pi)$, which can be inverse exponential in the input size [AV97; FR21]. Moreover, the Kempf–Ness function is usually not *strongly convex*, so one should not expect to get a poly $\log(1/\varepsilon)$ -convergence rate from simple gradient descent algorithms.

Box-constrained Newton methods. More recently, *box-constrained Newton methods* have made an appearance in the literature [CMTV17; ALOW17; AGL+18; BFG+19; CKV20]. This is a method for minimizing a certain class of convex functions where every iteration is essentially a Newton step, but constrained to a subdomain (box or ball) of essentially fixed size. The number of required iterations for approximately minimizing the objective is polynomial in a diameter bound R on an (approximate) solution, $\log \frac{1}{\delta}$ and a parameter known as the *robustness parameter* of the objective. Robustness of the objective guarantees that the local quadratic approximation obtained from a second-order Taylor expansion is relatively accurate; the size of the robustness parameter determines the diameter of the region in which such an approximation holds.

The Kempf–Ness function can be shown to be robust [BFG+19], with robustness parameter controlled by the weight norm $N(\pi)$ of the representation. This was used in the non-commutative setting to give polynomial time algorithms for operator scaling and the related orbit closure intersection problem [AGL+18], improving upon the results of [GGOW16].¹

A general non-commutative version of this second-order method is given in [BFG+19, Prop. 5.5, Thm. 5.7], and its guarantees are as follows:

Theorem 1.3.2 (Box-constrained Newton method). *Given semistable $v \in V \setminus \{0\}$ and $\delta > 0$, there exists an algorithm that solves the norm minimization problem in $\tilde{O}(R \text{ poly}(N(\pi), C, \log(1/\delta)))$ iterations. An iteration consists of computing an explicit gradient and Hessian, solving a Euclidean convex quadratic optimization problem, and basic linear algebraic operations. Here, R is a bound on the distance to an δ -approximate minimizer, $C = F_v(I) - F_v^*$ is the potential gap, and $\tilde{O}(\cdot)$ hides polylogarithmic terms in R .*

To make use of this guarantee, one has to bound the quantities R and C . The quantity C is usually bounded by making use of the structure of the invariant polynomials on the representation (although in the commutative case, more concrete bounds can be obtained, see Chapter 5). The best general bounds on R are linear in terms of the inverse weight margin $1/\gamma(\pi)$, which is exponentially large in general [BFG+19, Prop. 5.6].

In fact, the box-constrained Newton methods are known to be fundamentally incapable of providing polynomial-time algorithms for general scaling problems. The reason is that the distance to an approximate minimizer is in general exponential in the input size [FR21] (even in the commutative setting), and every iteration of a box-constrained Newton method is only capable of traversing an (almost) constant distance. Therefore one cannot always achieve optimality within a polynomial number of iterations.

¹We note that in the setting of operator scaling, solving the OCI problem, the null-cone problem and determining the non-commutative rank can be done efficiently through other approaches as well, see [GGOW16; IQS17; IQS18; DM20a; HH21; FSG23].

1. Introduction

In the commutative setting, the issue of large diameter bounds can be overcome. For example, one can appeal to the ellipsoid method to show that unconstrained geometric programs are solvable in polynomial time [NR99]. In the non-commutative setting no analog of the ellipsoid method is available, however, so one is forced to look for other methods.

Geometric obstructions. A significant obstruction to providing efficient algorithms is that the geometry of the domains makes it fundamentally more difficult to solve optimization problems. This is caused by the fact that they have *non-positive curvature*, whereas Euclidean space has zero curvature. As an example, the natural metric on the space $SU(2)\backslash SL(2, \mathbb{C})$ turns it into (3-dimensional) *hyperbolic space*, which has *constant* negative curvature; see Fig. 1.1 for an illustration of the Poincaré disk model of 2-dimensional hyperbolic space. Rusciano [Rus19] gave a (non-constructive) cutting-plane method in non-positive curvature, with a logarithmic dependence on the volume of the domain. Unfortunately, the volume of balls in manifolds of non-positive curvature can grow *exponentially* with the radius (even in constant dimension); in particular, this is the case for symmetric spaces of non-positive curvature, see e.g. [GN99]. This immediately suggests that a generalization of cutting-plane and/or ellipsoid methods to non-positive curvature should not suffice for solving scaling problems, assuming their runtime will, as in the Euclidean setting, depend logarithmically on the volume of a bounding ball, which would still be exponential here. In fact, it remains open whether there exists a first-order algorithm for minimizing Lipschitz geodesically convex functions, with polynomial dependence on the dimension, a diameter bound and a logarithmic dependence on the precision [CMB23] (in light of the exponential volume scaling of balls, this would be similar to the ellipsoid method in Euclidean space).

The exponential volume scaling can also be used to prove lower bounds in a black-box setting: there exist (natural) optimization problems for which, if one can only make queries to a function- and gradient oracle, any algorithm that finds an approximate minimizer must make a number of queries that is *linear* in the distance to the approximate minimizer [HM21a; CB22; CB23]. This again suggests that efficient algorithms for geodesic convex optimization in non-positive curvature in general, and for non-commutative optimization problems in particular, must make use of additional structure beyond diameter bounds, as the distance to an approximate minimizer is in general exponential in the input size [FR21].

1.4. Summary of results

This thesis is naturally divided into three parts, and here we discuss them in the order in which they appear.

Part I: Scaling problems and applications. We start by setting the mathematical stage in Chapter 2, with basic background on geometric invariant theory (with additional introductory material on algebraic geometry and algebraic groups), a proof of the Kempf–Ness theorem, properties of the moment map and moment polytopes, and a formal definition of the norm minimization and scaling problems.

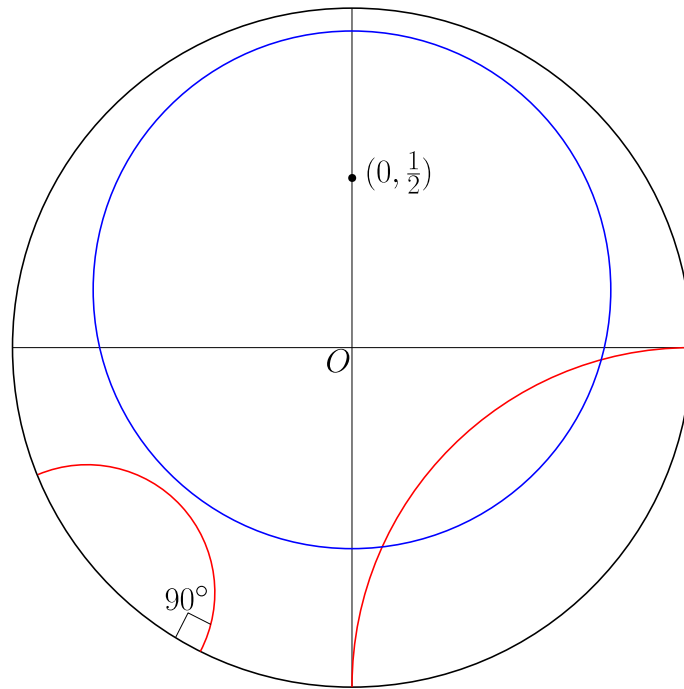


Figure 1.1.: A picture of the Poincaré disk model of hyperbolic space. The distance between two points $z, w \in \mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$ is given by $d(z, w) = \text{arcosh}(1 + 2|z - w|^2 / ((1 - |z|^2)(1 - |w|^2)))$. The horizontal and vertical axes and red arcs are geodesics, given by circular arcs meeting the boundary orthogonally. The blue curve indicates the boundary of a ball of radius 2 about the point $(0, \frac{1}{2})$.

Next, in Chapter 3, we identify a novel application of geometric invariant theory and algorithms for scaling problems in the context of quantum many-body physics. We use it to define a *minimal canonical form* for tensor networks defining projected entangled-pair states (PEPS). These are a higher-dimensional generalization of the well-known matrix product states, a class of computationally useful ground-state Ansätze for local Hamiltonians. This shows that previous undecidability results for testing properties of PEPS can be circumvented.

On a technical level, given a tensor $T \in (\mathbb{C}^{D_1 \times D_1} \otimes \dots \otimes \mathbb{C}^{D_m \times D_m})^d$, one can define a quantum state $|\Gamma\rangle$ on any *contraction graph* Γ , and these states are what are known as the PEPS. The resulting quantum state $|\Gamma\rangle$ is invariant under the simultaneous conjugation action of $GL(D_1) \times \dots \times GL(D_m)$ on T . To understand the physical properties of these quantum states $|\Gamma\rangle$, it is desirable to have canonical representatives of the orbit (closure) of T . We define such a canonical form T_{\min} , called the *minimal canonical form*: it is a minimum norm vector in the orbit closure (or equivalently, the solution to a scaling problem), and has excellent structural properties as a result of the general theory as explained in Section 1.2, in particular the theorems of Mumford and Kempf–Ness. Computing the minimal canonical form can be done with any of the algorithms for norm minimization, albeit not efficiently in all parameters, because the weight margin of the representation is inverse exponentially small in the bond dimensions D_1, \dots, D_m (unless $m = 1$).

Moreover, we show that two tensors T, S have a common minimal canonical form if and only if on any contraction graph Γ , $|\Gamma\rangle = |S_\Gamma\rangle$. This is achieved by

1. Introduction

observing that the coefficients of such states form a generating set for the invariant polynomials on the underlying representation, and can be seen as a generalization of well-known results from classical invariant theory [Pro76; Raz74; For86] on the invariant polynomials of simultaneous matrix conjugation for $GL(D)$. We also bound the size of the graphs Γ for which one has to check this condition, by relying on more modern tools from constructive invariant theory [Der00].

Part II: Interior-point methods for scaling. This part is concerned with the development of new classical algorithms for scaling problems. As mentioned previously, in the commutative case there are various approaches to obtaining efficient algorithms for commutative scaling problems, even when the approximate solution can be far away from the starting point. Chapters 4 and 5 are concerned with one such approach, namely to use the framework of (Euclidean) *interior-point methods* (IPMs) for unconstrained geometric programming problems. These form an essential part of the modern optimization toolbox, as they give polynomial iteration-complexity guarantees in great generality and in some cases state-of-the-art methods (such as for linear programming), and are also extremely performant in practice. We provide a gentle introduction to the theory of IPMs in Chapter 4.

Next in Chapter 5, we apply the IPM framework in the context of unconstrained geometric programming. We show that one can formulate IPMs whose complexity can be analyzed in terms of *condition numbers* that are defined in terms of the geometry of the moment polytope. For rational inputs, these condition numbers are bounded in terms of the input size, leading to polynomial iteration complexity bounds for unconstrained geometric programming and hence also for scaling problems:

Theorem 1.4.1 (IPM for commutative norm minimization). *Let $G = (\mathbb{C}^\times)^n$ and let $\pi: G \rightarrow GL(V)$ be a regular representation, given explicitly in terms of its weights $\Omega \subset \mathbb{Z}^n$. Given semistable $v \in V$ and $\delta > 0$, there exists an interior-point method which outputs $g_\delta \in G$ such that $F_v(g_\delta) \leq F_v^* + \delta$ within $O(\text{poly}(\log(1/\delta), \text{input size}))$ iterations. An iteration consists of computing an explicit gradient and Hessian, and basic linear algebraic operations.*

To go from commutative to non-commutative scaling problems, we next generalize the Euclidean interior-point method framework to the setting of Riemannian manifolds. An overview of the key ingredients and applications of this generalization is given in Chapter 7. For convenience, preliminaries on Riemannian geometry and geodesic convexity are collected in Chapter 6. In summary, the main achievements are as follows:

In Chapter 8, the Euclidean framework as discussed in Chapter 5 is extended to the Riemannian setting. We define an appropriate notion of self-concordance in the Riemannian setting. Essentially the same guarantees as in the Euclidean setting are obtained; of particular note are the quadratic convergence rate of Newton's method for self-concordant functions, and path-following methods for objectives on domains admitting a self-concordant barrier. More precisely, we prove the following:

Theorem 1.4.2 (Path-following method). *Let $D \subseteq M$ be an open, bounded, and convex domain in a complete Riemannian manifold M , and let $f, F: D \rightarrow \mathbb{R}$ be smooth convex*

functions, such that F is a self-concordant barrier with barrier parameter $\theta \geq 0$ and f has a closed convex extension. Let $\alpha > 0$ be such that $F_t := tf + F$ is α -self-concordant for all $t \geq 0$. Let $p \in D$ be sufficiently close to the analytic center of F , and let $\varepsilon > 0$. Then, using

$$O\left(\left(1 + \sqrt{\frac{\theta}{\alpha}}\right) \log\left(\frac{(\theta + \alpha)\|df_p\|_{F,p}^*}{\varepsilon\sqrt{\alpha}}\right)\right)$$

Newton iterations, one can find a point $p_\varepsilon \in D$ such that

$$f(p_\varepsilon) - \inf_{q \in D} f(q) \leq \varepsilon.$$

Of course, the above is not useful without explicit examples of self-concordant functions. In Chapter 9 we show that every symmetric space of non-positive curvature admits self-concordant functions, namely the squared distance to a point:

Theorem 1.4.3. *Let M be a symmetric space of non-positive curvature. Then for every $p_0 \in M$, the function $f: M \rightarrow \mathbb{R}$ given by $f(p) = d(p, p_0)^2$ is α -self-concordant for some $\alpha > 0$ that depends only on M .*

We use this to construct a self-concordant barrier for the manifold analogue of second-order cones (or rather a bounded version thereof).

In Chapter 10 we show that the IPM framework captures scaling problems as well as other natural geometric problems on non-positively curved symmetric spaces. In particular, we obtain algorithms for non-commutative scaling problems whose guarantees match the state-of-the-art as in Theorem 1.3.2:

Theorem 1.4.4 (IPM for norm minimization). *Let G be a connected reductive linear algebraic group and $\pi: G \rightarrow \mathrm{GL}(V)$ a regular representation. Given semistable $v \in V \setminus \{0\}$ and $\delta > 0$, there exists an interior-point method that solves the norm minimization problem in $\tilde{O}(R \cdot \mathrm{poly}(N(\pi), C, \log(1/\delta)))$ iterations. An iteration consists of computing an explicit gradient and Hessian, and basic linear algebraic operations. Here, R is a bound on the distance to a δ -approximate minimizer, $C = F_v(I) - F_v^*$ is the potential gap, and $\tilde{O}(\cdot)$ hides polylogarithmic terms in R .*

Showing that scaling problems are captured by the framework involves proving new estimates on the derivatives of the Kempf–Ness function, generalizing the robustness bounds that were essential to the box-constrained Newton methods discussed earlier. Moreover, computing geometric medians and minimum-enclosing balls can be solved to high precision using the IPM framework, whereas previous methods were only capable of efficiently providing low-precision solutions.

Part III: Quantum algorithms and lower bounds for scaling. In this part of the thesis, we explore the potential of quantum computers to provide faster algorithms for scaling problems than the classical state-of-the-art. We focus on improvements to basic quantum subroutines and the well-studied matrix scaling and balancing problems. In Chapter 11 we provide a short introduction to quantum computing, recall some basic quantum subroutines that we invoke later, and recall techniques for proving (query) lower bounds for quantum algorithms.

1. Introduction

Chapter 12 is then concerned with two basic problems. The first is: given quantum query access to a bit string $x \in \{0, 1\}^n$, find all $i \in [n]$ such that $x_i = 1$. This is a generalization of the *unstructured database search* problem, where one has to find a single index i such that $x_i = 1$. The search problem famously admits a quantum algorithm known as Grover's algorithm [Gro96] solving it with probability $\geq 2/3$ in time $\tilde{O}(\sqrt{n})$, whereas any classical algorithm must make $\Omega(n)$ queries to x to solve the problem with constant probability. Our contribution here is the following:

Theorem 1.4.5. *Let $x \in \{0, 1\}^n$ have Hamming weight $k = |x|$. Then there exists a quantum algorithm which, with probability $\geq 2/3$, finds all k marked indices using $O(\sqrt{nk})$ queries and $\tilde{O}(\sqrt{nk})$ other basic operations, while using only a small quantum memory.*

The query complexity above is optimal, as can be deduced from lower bounds for threshold functions [BBC+01]. Previous algorithms either used a factor $\log(k)$ more queries, or a factor k more basic operations. For simplicity, the above is stated for solving the problem with constant success probability, but we note that one can achieve a high success probability with a better complexity than obtained from a standard boosting procedure.

The second problem is: given $\varepsilon > 0$ and quantum query access to a vector $v \in [0, 1]^n$, compute a $(1 \pm \varepsilon)$ -multiplicative approximation of the sum $\sum_{i=1}^n v_i$. Previous approaches solved this problem in $O(\sqrt{n}/\varepsilon)$ queries and a similar number of other operations [Gro97; Gro98; BHMT02]. We improve this as follows:

Theorem 1.4.6. *Let $v \in [0, 1]^n$ and $\varepsilon > 0$. Then there exists a quantum algorithm which finds, with probability $\geq 2/3$, a $(1 \pm \varepsilon)$ -multiplicative approximation of $\sum_{i=1}^n v_i$, using $O(\sqrt{n}/\varepsilon)$ queries and $\tilde{O}(\sqrt{n}/\varepsilon)$ other basic operations.*

Again, the above is only stated for constant success probability, but improvements are possible for higher success probability.

Next, we turn to quantum algorithms for matrix scaling and balancing, and the limitations of using quantum computers for these problems. In Chapter 13 we give a more comprehensive overview of the literature on these problems and state the main results of Chapters 14 to 17 more precisely. These results are summarized in Fig. 1.2. In this chapter we also define our input model, set our notation for the following chapters, and build several relevant quantum subroutines for later use. This includes quantum subroutines for computing logarithms of sums of exponentials, and testing whether a matrix is approximately scaled. For these subroutines, we rely on the improved summation technique from Chapter 12.

In Chapter 14 we discuss quantum implementations of Sinkhorn's algorithm for matrix scaling, including a version with random updates; the latter analysis extends naturally to Osborne's algorithm for matrix balancing. For matrix scaling, this achieves the following:

Theorem 1.4.7. *Given an $n \times n$ matrix A with non-negative entries and probability distributions $r, c \in \mathbb{R}_{>0}^n$, if A can be (asymptotically) (r, c) -scaled, then scaling matrices X and Y such that $\|r(XAY) - r\|_1 + \|c(XAY) - c\|_1 \leq \varepsilon$ can be found using $\tilde{O}(n^{1.5}/\varepsilon^3)$ quantum queries to the entries of A , and a similar number of other operations. When A is entrywise-positive, this bound can be improved to $\tilde{O}(n^{1.5}/\varepsilon^2)$ quantum queries and a similar number of other operations.*

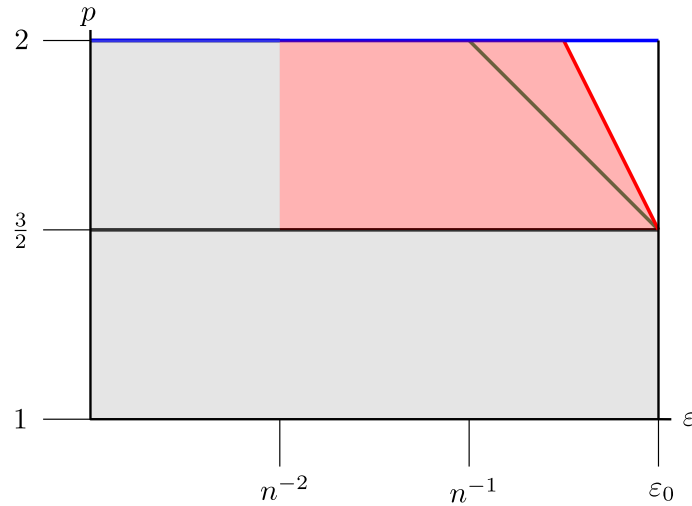


Figure 1.2.: A schematic representation of the complexity and limitations of quantum algorithms for matrix scaling of entrywise-positive matrices. The horizontal axis gives the precision ε as a function of n (with ε_0 being a certain constant), whereas the vertical axis gives the exponent p in a complexity of order n^p for the problem of ε - ℓ^1 -scaling. The blue line represents the classical state-of-the-art complexity $\tilde{O}(n^2)$, the red line represents the quantum box-constrained Newton method with complexity $\tilde{O}(n^{1.5}/\varepsilon)$ (Theorem 1.4.8). The area shaded in red indicates the possible complexities a quantum algorithm for matrix scaling could have in the respective ε -regime, whereas the area shaded in grey is ruled out by Theorem 1.4.9. The green line corresponds to the lower bound $\Omega(n^{1.5}/\sqrt{\varepsilon})$ for the problem of computing ε - ℓ^1 -approximations of the vector of row sums of a normalized matrix (Theorem 17.7.1).

For comparison, the classical Sinkhorn algorithm would take time $\tilde{O}(n^2/\varepsilon^2)$, or $\tilde{O}(n^2/\varepsilon)$ in the entrywise-positive case. A similar statement holds for matrix balancing.

Chapter 15 describes our second-order quantum algorithms for scaling and balancing. These are based on the box-constrained Newton methods due to [CMTV17] and recent work on quantum algorithms for graph sparsification [AW22]. We prove the following:

Theorem 1.4.8. *Given an $n \times n$ matrix A with non-negative entries and probability distributions $r, c \in \mathbb{R}_{>0}^n$, if A can be (asymptotically) (r, c) -scaled, then scaling matrices X and Y such that $\|r(XAY) - r\|_1 + \|c(XAY) - c\|_1 \leq \varepsilon$ can be found using $\tilde{O}(R^{1.5} n^{1.5}/\varepsilon)$ quantum queries to the entries of A , and a similar number of other operations. Here, R is a bound on the distance to an $O(\varepsilon^2)$ -approximate minimizer.*

For matrices A which are entrywise-positive, $R = \tilde{O}(1)$ and hence the complexity is $\tilde{O}(n^{1.5}/\varepsilon)$.

The classical equivalent of this algorithm would find an ε - ℓ^1 -scaling in time $\tilde{O}(n^2)$ for entrywise-positive matrices. A similar result holds for matrix balancing, although there one only obtains ε - ℓ^2 -balancings (as opposed to ε - ℓ^1) for technical reasons.

1. Introduction

Lastly, in Chapters 16 and 17, we prove quantum query lower bounds in the constant- ε and small- ε regimes, respectively. These are summarized as follows:

Theorem 1.4.9. *There exists a constant $\varepsilon_0 > 0$ such that ε_0 - ℓ^1 -scaling a matrix A to uniform marginals requires $\Omega(n^{1.5})$ queries to A . Moreover, for an explicit $\varepsilon = 1/\text{poly}(n)$, ε - ℓ^1 -scaling A to uniform marginals requires $\tilde{\Omega}(n^2)$ queries to A .*

The lower bound bound in the constant-precision regime is based on a reduction to a variant of multiple search problems. The lower bound in the high-precision regime is based on a reduction to certain counting problems, and heavily relies on properties of the convex potential from Eq. (1.1.1), along with a concentration argument. The $\tilde{\Omega}(n^2)$ lower bound implies that essentially every entry of A must be queried, and no general improvement over the classical state of the art is possible in this regime, since for entrywise-positive matrices classical algorithms can find ε - ℓ^1 -scalings in time $\tilde{O}(n^2)$. We also prove a $\Omega(n^{1.5}/\sqrt{\varepsilon})$ -lower bound for finding ε - ℓ^1 -approximations of the vector of row sums of matrix in Theorem 17.7.1. This is *morally* also a lower bound for scaling: all algorithms (to the best of our knowledge) explicitly compute (an approximation of) the row and column sums of the matrix, and an ε - ℓ^1 -approximation of the row and column sums is exactly enough to test whether a matrix is ε - ℓ^1 -scaled.

Part I.

Scaling problems and applications

2. Setting the stage

In this chapter, we set the mathematical stage for the scaling problems as discussed in Chapter 1. First in Section 2.1 we set some basic notation. Next, Section 2.2 collects some preliminary material on algebraic varieties and algebraic groups. Section 2.3 then turns to the topic of geometric invariant theory, where we discuss Mumford's theorem (Theorem 2.3.7), properties of the ring of invariant polynomials, the Hilbert–Mumford theorem (Theorem 2.3.16), and the notion of stability and its relation to quotients of projective varieties. This is followed by a proof of the Kempf–Ness theorem in Section 2.4 and a detailed discussion of the moment map in Section 2.5. Lastly, we formally define the computational problems (scaling and norm minimization) in detail in Section 2.6, as well as a quantitative relation between these.

2.1. Notation and conventions

Before we get to the main content of this chapter, we fix some terminology and notation. For $n \geq 1$, we write $[n]$ for the set $\{1, \dots, n\}$. For $1 \leq p \leq \infty$, we write $\|\cdot\|_p$ for the p -norm on \mathbb{R}^n or \mathbb{C}^n . The standard inner product on \mathbb{R}^n or \mathbb{C}^n is defined by $\langle u, v \rangle = u^*v = \sum_{i=1}^n \overline{u_i}v_i$. In particular, our inner products are complex-linear in the *second* argument, to be consistent with physicists' Dirac notation.

The space of $m \times n$ matrices over a field \mathbb{F} is denoted by $\mathbb{F}^{m \times n}$, or as $\text{Mat}_{m \times n}(\mathbb{F})$ when additional superscripts are necessary. We write $\text{Tr}[A] = \sum_{i=1}^n A_{ii}$ for the trace of a matrix $A \in \mathbb{F}^{n \times n}$. For $\mathbb{F} \in \{\mathbb{R}, \mathbb{C}\}$, the Hilbert–Schmidt inner product on $\mathbb{F}^{m \times n}$ is defined by $\langle A, B \rangle = \text{Tr}[A^*B]$, where A^* is the conjugate transpose of A . The induced norm is denoted by $\|A\|_{\text{HS}} = \sqrt{\langle A, A \rangle}$. We also write $\text{Herm}(n) \subseteq \mathbb{C}^{n \times n}$ for the Hermitian (i.e., self-adjoint) matrices. The operator norm of $A \in \mathbb{F}^{m \times n}$ is defined by $\|A\|_{\infty} = \sup_{\|u\|_2=1} \|Au\|_2$.

The group of invertible $n \times n$ matrices over a field \mathbb{F} is denoted by $\text{GL}(n, \mathbb{F})$, and $\text{SL}(n, \mathbb{F})$ consists of those $g \in \text{GL}(n, \mathbb{F})$ with $\det(g) = 1$, where \det is the determinant. The special case $\text{GL}(1, \mathbb{F})$ is denoted by $\mathbb{F}^\times = \mathbb{F} \setminus \{0\}$. We also write $\text{U}(n) = \{g \in \text{GL}(n, \mathbb{C}) : g^* = g^{-1}\}$ for the unitary matrices over \mathbb{C} .

2.2. Preliminaries on algebra, geometry and groups

In this section we collect preliminaries on affine and algebraic varieties, and linear algebraic groups. The reader is encouraged to consult this section only as necessary.

This chapter is partially adapted from [AMN+23].

2.2.1. Affine varieties

For $m \geq 1$, we let $\mathbb{C}[x_1, \dots, x_m]$ denote the ring of polynomials in m variables x_1, \dots, x_m with coefficients in \mathbb{C} . For given polynomials $p_1, \dots, p_r \in \mathbb{C}[x_1, \dots, x_m]$, the (embedded) *affine variety* $V(p_1, \dots, p_r) \subseteq \mathbb{C}^m$ defined by the p_j is their common zero set,¹ i.e.,

$$V(p_1, \dots, p_r) = \{(x_1, \dots, x_m) \in \mathbb{C}^m : p_j(x_1, \dots, x_m) = 0, \forall j \in [r]\}.$$

For an ideal $J \subseteq \mathbb{C}[x_1, \dots, x_m]$, we may similarly define $V(J)$ as the common zero set of all polynomials $f \in J$.

Affine varieties in \mathbb{C}^m have the following properties:

- (i) $\mathbb{C}^m = V(0)$ is an affine variety, and so is $\emptyset = V(\mathbb{C}[x_1, \dots, x_m])$.
- (ii) If X_1, \dots, X_l are affine varieties, then so is $X_1 \cup \dots \cup X_l$.
- (iii) If $\{X_i\}_{i \in I}$ is a (possibly infinite) collection of affine varieties, then $\bigcap_{i \in I} X_i$ is an affine variety.

These three properties imply that if we declare the affine varieties to be the *closed* subsets of \mathbb{C}^m , then this forms a topology, known as the Zariski topology. Any polynomial $p \in \mathbb{C}[x_1, \dots, x_m]$ viewed as a function $\mathbb{C}^m \rightarrow \mathbb{C}$ is then a continuous map (with respect to the Zariski topologies on \mathbb{C}^m and \mathbb{C} ; we recall that continuity means preimages of open sets are open, or equivalently preimages of closed sets are closed). Notably, by the fundamental theorem of algebra and the fact that $\mathbb{C}[x]$ is a principal ideal domain, the closed subsets of \mathbb{C} are \mathbb{C} itself and the finite subsets.²

Since we may view $p \in \mathbb{C}[x_1, \dots, x_m]$ as functions $\mathbb{C}^m \rightarrow \mathbb{C}$, one may consider the restriction $p|_X$ to an affine variety $X \subseteq \mathbb{C}^m$; this is automatically a continuous function $X \rightarrow \mathbb{C}$ with respect to the subspace topology. The restriction map is an algebra³ homomorphism $\mathbb{C}[x_1, \dots, x_m] \rightarrow \{X \rightarrow \mathbb{C} \text{ continuous}\}$, and its kernel (i.e., the polynomials which restrict to the zero function) is called the *vanishing ideal* $I(X)$ of X . By the first isomorphism theorem, the quotient $\mathbb{C}[x_1, \dots, x_m]/I(X)$ is isomorphic to the image of the restriction map, which forms a subalgebra of $\{X \rightarrow \mathbb{C} \text{ continuous}\}$. We shall refer to this subalgebra as the *coordinate ring* or *ring of regular functions* of X , and denote it by $\mathbb{C}[X]$.

More generally, if (X, R) is a pair where X is a topological space, and R is a subalgebra of the algebra of continuous functions $X \rightarrow \mathbb{C}$, then we call this pair an affine variety if there exists some $m \geq 0$ and a homeomorphism $f: X \rightarrow Z$ with $Z \subseteq \mathbb{C}^m$ Zariski-closed, such that the map $f^*: \mathbb{C}[Z] \rightarrow R$ given by $f^*(p) = p \circ f$ is an isomorphism of algebras. Again, we refer to the algebra R as the *coordinate ring* or *ring of regular functions* of X .

¹It may happen that this set is empty, but only if the ideal in $\mathbb{C}[x_1, \dots, x_m]$ generated by the p_j is all of $\mathbb{C}[x_1, \dots, x_m]$.

²As a word of caution, a continuous function need not be a polynomial: every bijection $\mathbb{C} \rightarrow \mathbb{C}$ is continuous with respect to the Zariski topology. In fact, there are $2^{|\mathbb{C}|}$ many such bijections and only $|\mathbb{C}|$ many polynomials (since polynomials have finitely many non-zero coefficients).

³An algebra over \mathbb{C} is a vector space over \mathbb{C} endowed with a \mathbb{C} -bilinear multiplication map. We assume that all algebras are associative, unital, and commutative unless stated otherwise.

Although not obvious from the above definition, it turns out that X is uniquely determined (as a topological space) by the algebra R . The algebras R which can arise as coordinate rings of affine varieties can also be characterized as follows:

- It is *finitely generated*, in the sense that there exist $q_1, \dots, q_r \in R$ such that every element $q \in R$ can be written as a polynomial expression in the q_j (and the constant function 1).
- Its *nilradical*, consisting of those $p \in R$ such that $p^n = 0$ for some $n \geq 0$, is the zero ideal $\{0\} \subseteq R$.

To see that these hold for coordinate rings of affine varieties, it suffices to check for R of the form $\mathbb{C}[Z]$ with $Z \subseteq \mathbb{C}^m$ Zariski-closed, where these properties are obvious: $x_1|_Z, \dots, x_m|_Z$ generate the algebra, and if p^n is a polynomial which vanishes on Z , then so is p . That all these algebras arise as coordinate rings of affine varieties follows from Hilbert's Nullstellensatz.

Theorem 2.2.1 (Hilbert's Nullstellensatz, [Wal17, Thm. 1.3]). *Let $J \subseteq \mathbb{C}[x_1, \dots, x_m]$ be an ideal. If J is a proper ideal, then the vanishing locus $V(J) = \{(x_1, \dots, x_m) \in \mathbb{C}^m : p(x_1, \dots, x_m) = 0, p \in J\}$ of J is non-empty. Furthermore, the set $I(V(J))$ of polynomials $p \in \mathbb{C}[x_1, \dots, x_m]$ vanishing on $V(J)$ is the radical $\sqrt{J} = \{p \in \mathbb{C}[x_1, \dots, x_m] : p^n \in J \text{ for some } n \geq 1\}$ of J .*

This can be used to show that every R which has trivial nilradical and is finitely generated with generators q_1, \dots, q_m arises as the coordinate ring of an affine variety. The algebra homomorphism $\varphi: \mathbb{C}[x_1, \dots, x_m] \rightarrow R$, given by $x_j \mapsto q_j$ is surjective. By the first isomorphism theorem, R is isomorphic to $\mathbb{C}[x_1, \dots, x_m]/\ker(\varphi)$. As R has trivial nilradical, $\ker(\varphi)$ must be a radical ideal, in the sense that if $p^n \in \ker(\varphi)$ for some $n \geq 1$, then $p \in \ker(\varphi)$. Therefore the Nullstellensatz gives $I(V(\ker(\varphi))) = \sqrt{\ker(\varphi)} = \ker(\varphi)$, and $R \cong \mathbb{C}[Z]$ where $Z = V(\ker(\varphi))$.

As important as the affine varieties themselves, if not more, are maps between them. For two affine varieties (X, R) and (Y, S) , we say that a continuous map $f: X \rightarrow Y$ is a *regular map* or *morphism of affine varieties* if $f^*S \subseteq R$ (recall that R and S are assumed to be subalgebras of the ring of continuous \mathbb{C} -valued functions on X, Y , and $f^*S = \{p \circ f : p \in S\}$).

Given two affine varieties (X, R) and (Y, S) , we define their *product* to be the affine variety whose underlying set is $X \times Y$, and whose coordinate ring is $R \otimes S$, viewed as a subalgebra of the \mathbb{C} -valued functions on $X \times Y$. Note that $R \otimes S$ is finitely generated because R and S are. However, we endow $X \times Y$ with the Zariski topology, rather than the product topology. This topology is defined such that if we choose embeddings $X \subseteq \mathbb{C}^n$ and $Y \subseteq \mathbb{C}^m$ as Zariski-closed subsets, then the topology on $X \times Y$ is the subspace topology on $X \times Y$ with respect to the Zariski topology on $\mathbb{C}^n \times \mathbb{C}^m = \mathbb{C}^{n+m}$.

2.2.2. Algebraic varieties

While we shall forego a precise definition, it is convenient to enlarge the category of spaces to that of *algebraic varieties*. The idea is that although the category of affine varieties is well-behaved, it does not allow us to do all the geometry we might

2. Setting the stage

wish to. For instance, the projective space \mathbb{P}^m , whose points are lines through the origin in \mathbb{C}^{m+1} , is a classical object of interest in algebraic geometry, but it is not an affine variety. One way to remedy this is as follows. Instead of studying a topological space X with an algebra of functions R that are *globally* defined on X , the idea is to specify for every open subset $U \subseteq X$ a “coordinate ring” $\mathcal{O}_X(U)$, which is again a subalgebra of continuous functions $U \rightarrow \mathbb{C}$. We impose certain compatibility conditions on \mathcal{O}_X (it must be a *sheaf*), the pair (X, \mathcal{O}_X) is called a *ringed space*, and \mathcal{O}_X is called the *sheaf of regular functions* or the *structure sheaf*.⁴ An algebraic pre-variety is then a ringed space which is covered by a collection of open subsets $\{U_i\}_{i \in I} \subseteq X$ such that $(U_i, \mathcal{O}_X(U_i))$ is an affine variety. An *algebraic variety* is then a *separated* algebraic pre-variety, which means that the diagonal $\Delta_X \subset X \times X$ is closed (with respect to the Zariski topology on $X \times X$). A *regular map* or *morphism* between algebraic varieties X and Y is then a function $f: X \rightarrow Y$ such that if $U \subseteq Y$ is open and $p \in \mathcal{O}_Y(U)$ is a regular function on U , then $f^*(p) = p \circ f$ is a regular function on $f^{-1}(U)$, i.e., $p \circ f \in \mathcal{O}_X(f^{-1}(U))$.

For an affine variety X , the sheaf of regular functions is determined as follows. Every open set $U \subseteq X$ is the union of basic open sets $D_p = X \setminus V(p) \subseteq X$ where $p \in \mathbb{C}[X]$ (recall that $V(p)$ is the vanishing set of p). The regular functions on U are then those $f: U \rightarrow \mathbb{C}$ such that for every $x \in U$, there exists $p \in \mathbb{C}[X]$ such that $x \subseteq D_p \subseteq U$ and $f|_{D_p}$ is the restriction of some $\mathbb{C}[X][p^{-1}]$, i.e., $f(y) = \sum_{j=0}^k q_j(y)p(y)^{-j}$ for all $y \in D_p$ and some $q_0, \dots, q_k \in \mathbb{C}[X]$.

As an example, consider $X = \mathbb{C}^2$ with coordinates x_1, x_2 and $U = X \setminus \{x_1 x_2 = 0\}$. Then the regular functions on U are those $f: U \rightarrow \mathbb{C}$ are those in $\mathbb{C}[x_1, x_2, (x_1 x_2)^{-1}]$; for instance, $f(x_1, x_2) = (x_1 + x_2)/(x_1 x_2)$. On the other hand, the regular functions on $X \setminus \{0\}$ are just restrictions of regular functions on X : it would have to be simultaneously in $\mathbb{C}[x_1, x_2, x_2^{-1}]$ and $\mathbb{C}[x_1, x_2, x_1^{-1}]$, i.e., in $\mathbb{C}[x_1, x_2] = \mathbb{C}[X]$.⁵

The canonical example of an algebraic variety that is not affine is projective space \mathbb{P}^m . As a set, it is given by $\mathbb{P}^m = (\mathbb{C}^{m+1} \setminus \{0\})/\sim$ where $x \sim y$ if and only if there exists $\lambda \in \mathbb{C}^\times = \mathbb{C} \setminus \{0\}$ such that $y = \lambda x$. A typical element of \mathbb{P}^m is denoted by $[x_0 : \dots : x_m]$, the equivalence class of $x = (x_0, \dots, x_m) \in \mathbb{C}^{m+1} \setminus \{0\}$. For a collection $S \subseteq \mathbb{C}[x_0, \dots, x_m]$ of *homogeneous* polynomials, the set of $[x_0 : \dots : x_m]$ such that $p(x_0, \dots, x_m) = 0$ for all $p \in S$ is well-defined. The Zariski topology on \mathbb{P}^m is then defined by declaring such common zero sets to be the closed subsets.

The most important open subsets of \mathbb{P}^m are those given by the inequality $x_i \neq 0$, where $i \in \{0, \dots, m\}$. On this open subset, the regular functions are those of the form $[x_0 : \dots : x_m] \mapsto q(\frac{x_0}{x_i}, \dots, \widehat{\frac{x_i}{x_i}}, \dots, \frac{x_m}{x_i})$ where $q \in \mathbb{C}[z_1, \dots, z_m]$ is a polynomial, and the hat indicates omission of that argument.

Lastly, we recall that a topological space X is *irreducible* if, whenever $X = X_0 \cup X_1$ for closed subsets $X_0, X_1 \subseteq X$, one has $X = X_0$ or $X = X_1$. For an affine variety X , irreducibility is equivalent to the coordinate ring $\mathbb{C}[X]$ being an integral domain,

⁴The tuple (X, \mathcal{O}_X) is in fact a *locally ringed space*, which means that the *stalk* at each point is a local ring, i.e., has a unique maximal ideal. The stalk at a point consists of “functions defined on an infinitesimal neighbourhood” of the point, and the maximal ideal is given by the functions vanishing at the point.

⁵This phenomenon occurs more generally in complex analysis: if $U \subseteq \mathbb{C}^m$ is a Euclidean open set and $A \subseteq U$ is an analytic set of codimension ≥ 2 , then any holomorphic function f on $U \setminus A$ extends uniquely to a holomorphic function on U [GH78, p. 7, p. 396].

i.e., having no non-zero zero divisors.⁶ The *Krull dimension* of topological space X is defined as the maximal length d of a chain $\emptyset \neq X_0 \subset \cdots \subset X_d \subseteq X$ of distinct closed irreducible subspaces; if no such subspaces exist, then the Krull dimension is defined to be $-\infty$. The Krull dimension of \mathbb{C}^n endowed with the Zariski topology is n . Note that the Krull dimension does not always capture the “usual” notion of dimension, since e.g. the only closed non-empty irreducible subsets of \mathbb{C}^n with the *standard* topology are singletons, and hence has dimension 0.

2.2.3. Linear algebraic groups and their representations

Important examples of algebraic varieties for us are the *linear algebraic groups*. An algebraic group is simply an algebraic variety G with a group structure that is compatible with the variety structure: a multiplication map $G \times G \rightarrow G$, an inversion map $G \rightarrow G$ and an identity element $e \in G$, which satisfy the group laws, and such that multiplication and inversion are regular maps.⁷

The canonical example of an algebraic group is $GL(n, \mathbb{C})$, consisting of all $n \times n$ invertible matrices with entries in \mathbb{C} . This is an affine variety, even though it is not Zariski-closed in $\mathbb{C}^{n \times n}$. Observe that $A \in \mathbb{C}^{n \times n}$ is in $GL(n, \mathbb{C})$ if and only if $\det(A) \neq 0$. To encode this as the vanishing set of a polynomial, we add an extra variable t and view $GL(n, \mathbb{C})$ as the set of those $(a_{11}, a_{12}, \dots, a_{nn}, t) \in \mathbb{C}^{n^2+1}$ such that $\det(A)t - 1 = 0$, where $A = (a_{ij})_{i,j=1}^n$. The multiplication on $GL(n, \mathbb{C})$ is of course given by matrix multiplication, which is a regular map since entries of the resulting matrix are polynomials in the two matrices; the inversion is also regular because $A^{-1} = \det(A)^{-1} \text{adj}(A)$ where $\text{adj}(A)$ is the adjugate matrix, whose entries are given by (signed) minors of A , hence also polynomials in the a_{ij} .

A *linear algebraic group* is a subgroup $G \subseteq GL(n, \mathbb{C})$ which is Zariski-closed in $GL(n, \mathbb{C})$, i.e., there are polynomials p_1, \dots, p_r in the matrix entries of g and $\det(g)^{-1}$ such that $G = \{g \in GL(n, \mathbb{C}) : p_j(g) = 0, j \in [r]\}$. Examples of interest are the special linear group $SL(n, \mathbb{C})$, consisting of those A with $\det(A) = 1$, the orthogonal and special orthogonal groups $O(n, \mathbb{C})$ and $SO(n, \mathbb{C})$, the symplectic group, but also the group of invertible upper triangular matrices. The group $\mathbb{C}^\times = \mathbb{C} \setminus \{0\} = GL(1, \mathbb{C})$ and products thereof are particularly simple⁸ examples as well. We shall refer to $(\mathbb{C}^\times)^d$ as an *algebraic torus* of dimension d (and extend this language to G which are isomorphic to $(\mathbb{C}^\times)^d$). A notable non-example of linear algebraic groups is the unitary group $U(n)$: its defining equations are $gg^* = I_n = g^*g$, which involves complex conjugation, and hence is not a polynomial in the entries of g .⁹

⁶More generally irreducibility corresponds to the nilradical being a prime ideal, but since our coordinate rings have trivial nilradical, this corresponds to the zero ideal being prime, i.e., $\mathbb{C}[X]$ being an integral domain.

⁷Note that $X \times X$ is again endowed with the Zariski topology.

⁸Unfortunately, this group is not *simple* in the sense of Lie groups, because that definition excludes abelian groups.

⁹More generally, if $G \subseteq GL(n, \mathbb{C})$ is a linear algebraic group, then G is either finite or unbounded (in the Euclidean sense). To see this, observe that G is affine, hence has finitely many irreducible components by the Lasker–Noether theorem. On each irreducible component, a regular function is either unbounded or constant: their image is irreducible and the irreducible subsets of \mathbb{C} are the singletons and \mathbb{C} itself. In particular every coordinate map $G \mapsto \mathbb{C}, g \mapsto g_{ij}$ is unbounded or constant, and if all of them are constant then G has finitely many elements.

2. Setting the stage

As is a common theme in mathematics, we attempt to reduce complicated questions to linear algebra problems. Recall that a *representation* is a group homomorphism $\pi: G \rightarrow GL(V)$, where V is a finite-dimensional vector space over \mathbb{C} . Such a representation is *regular* if the matrix entries of $\pi(g)$ with respect to any basis of V are polynomial functions of the matrix entries of g and of $\det(g)^{-1}$ (i.e., the components are regular functions on G). We call a representation π *irreducible* if every G -invariant subspace $W \subseteq V$ satisfies $W = \{0\}$ or $W = V$.¹⁰ Furthermore, a representation π is *completely reducible* if there exist irreducible subrepresentations $V_1, \dots, V_r \subseteq V$ such that $V = V_1 \oplus \dots \oplus V_r$. To make the theory (more) tractable, we further restrict the class of groups that we are interested in:

Definition 2.2.2 (Linearly reductive group). Let $G \subseteq GL(n, \mathbb{C})$ be a linear algebraic group. Then G is called *linearly reductive* if for every regular representation $\pi: G \rightarrow GL(V)$ and G -invariant subspace $V_1 \subseteq V$, there exists a G -invariant subspace $V_2 \subseteq V$ such that $V = V_1 \oplus V_2$.

By induction on the dimension of V , one can show that every regular representation of a linearly reductive group is completely reducible. When one studies representations of finite groups, or more generally compact Lie groups, every representation satisfies the above property, as there one can introduce an invariant inner product by an averaging procedure. However, for algebraic groups, the notion is non-trivial: for general G it may happen that although a representation V is not irreducible, there do not exist G -invariant subspaces $V_1, V_2 \subseteq V$ such that $V = V_1 \oplus V_2$ (and hence V is not completely reducible); we give the canonical example below.¹¹

Example 2.2.3. Consider the linear algebraic group

$$G = \left\{ \begin{bmatrix} 1 & z \\ 0 & 1 \end{bmatrix} : z \in \mathbb{C} \right\} \subseteq GL(2, \mathbb{C}),$$

which is isomorphic to \mathbb{C} . Then G acts by left-multiplication on $V = \mathbb{C}^2$. If $0 \subsetneq W \subsetneq V$ is a non-zero G -invariant proper subspace, then $\dim(W) = 1$ and $W = \text{span}(w)$ for some $0 \neq w \in W$. The G -invariance then implies that w must be a common eigenvector of every $g \in G$; but $\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ has only $e_1 \in \mathbb{C}^2$ as an eigenvector, hence $W = \text{span}(e_1)$. This is therefore the unique non-zero G -invariant proper subspace, and as a consequence there does not exist a G -invariant $W' \subseteq V$ such that $V = W \oplus W'$.

However, some respite is offered by the following theorem, which gives a fairly explicit characterization of the linearly reductive groups:

Theorem 2.2.4 ([Wal17, Lem. 3.5, Thm. 3.13]). *Let $G \subseteq GL(n, \mathbb{C})$ be a linear algebraic group. Assume that G is symmetric, that is, for every $g \in G$, its adjoint g^* is also in G . Then G is linearly reductive.*

¹⁰An irreducible π must be eaten in one biG bite.

¹¹In a sense, the example described is also the *only* obstruction to being linearly reductive (in characteristic 0). An equivalent definition of reductivity is that every smooth connected unipotent normal subgroup of G is trivial, where a unipotent group is one that is isomorphic to a subgroup of the upper triangular matrices with ones on the diagonal.

Moreover, if G is linearly reductive, then there exists some $g \in \mathrm{GL}(n, \mathbb{C})$ such that gGg^{-1} is symmetric.

The first part of this theorem is most important to us, and its proof relies on the interplay between the Zariski and Euclidean topology on $\mathrm{GL}(n, \mathbb{C})$. This requires some terminology from differential geometry. We recall that a *Lie group* is a topological group which is also a smooth manifold, such that the inversion and multiplication maps are smooth.

If $G \subseteq \mathrm{GL}(n, \mathbb{C})$ is a linear algebraic group, then G is Zariski-closed in $\mathrm{GL}(n, \mathbb{C})$. This implies that it is also closed subset with respect to the Euclidean topology on $\mathrm{GL}(n, \mathbb{C})$. Furthermore, with this topology, it admits a smooth atlas such that G is a Lie group, a result due to von Neumann [Neu29] and É. Cartan [Car52, Sec. II.III] (when the outer group is an abstract Lie group):

Theorem 2.2.5 (Closed subgroup theorem, [GW09, Prop. 1.3.12]). *Let $H \subseteq \mathrm{GL}(n, \mathbb{C})$ be a closed subgroup with respect to the Euclidean topology. Then H is a Lie group.*

This also leads to examples of Lie groups which are not algebraic. The most important for us is the unitary group $U(n)$, which consists of those $g \in \mathrm{GL}(n, \mathbb{C})$ such that $gg^* = I_n = g^*g$. This group is compact with respect to the Euclidean topology. Even though it is not linear algebraic, there is still a strong connection to linear algebraic groups:

Theorem 2.2.6. *Let $G \subseteq \mathrm{GL}(n, \mathbb{C})$ be a symmetric linear algebraic group. Then $K = G \cap U(n)$ is a maximal compact subgroup of G with respect to the Euclidean topology. Furthermore, the Zariski-closure of K in $\mathrm{GL}(n, \mathbb{C})$ is G .*

To relate this to the reductivity of G , we require one last ingredient:

Theorem 2.2.7. *Let $K \subseteq \mathrm{GL}(n, \mathbb{C})$ be a Lie group, compact with respect to the Euclidean topology. Then there exists a unique left Haar measure on K , i.e., a left- K -invariant Borel probability measure on K .*

Corollary 2.2.8. *Let $K \subseteq \mathrm{GL}(n, \mathbb{C})$ be a Lie group, compact with respect to the Euclidean topology. Let $\pi: K \rightarrow \mathrm{GL}(V)$ be a continuous representation of K . Then there exists an inner product $\langle \cdot, \cdot \rangle$ (complex linear in the second argument) on V such that $\pi(K) \subseteq U(V)$. Furthermore, if $V_0 \subseteq V$ is a K -invariant subspace, then there exists a K -invariant subspace $V_1 \subseteq V$ such that $V = V_0 \oplus V_1$.*

Proof. Take an arbitrary inner product $\langle \cdot, \cdot \rangle'$ on V , then define

$$\langle v, w \rangle = \int_K \langle k \cdot v, k \cdot w \rangle' dk,$$

where integration is performed with respect to the left Haar-measure on K . For any K -invariant subspace $V_0 \subseteq V$, the orthogonal complement $V_1 := V_0^\perp$ with respect to $\langle \cdot, \cdot \rangle$ is then K -invariant and satisfies $V = V_0 \oplus V_1$. \square

Corollary 2.2.9. *Let $G \subseteq \mathrm{GL}(n, \mathbb{C})$ be a symmetric linear algebraic group. Let $\pi: G \rightarrow \mathrm{GL}(V)$ be a regular representation of G , and let $V_0 \subseteq V$ be a G -invariant subspace. Then there exists a G -invariant subspace V_1 such that $V = V_0 \oplus V_1$. In particular, G is linearly reductive.*

2. Setting the stage

Proof. The subspace V_0 is also K -invariant, so there exists a decomposition $V = V_0 \oplus V_1$ where V_1 is K -invariant. The claim is that V_1 is also G -invariant. Let $\Pi_0: V \rightarrow V_0$ be the projection. Then for $v \in V_1$, the set $\{g \in G : \Pi_0(g \cdot v) = 0\}$ is defined by a polynomial system of equations (since Π_0 is just a linear map), and contains K ; but K is Zariski-dense in G , so it must be all of G . Therefore $\Pi_0(g \cdot v) = 0$ and hence $g \cdot v \in V_1$ for all $g \in G$. \square

Definition 2.2.10. For a Lie group $G \subseteq GL(n, \mathbb{C})$, we define its Lie algebra by

$$\text{Lie}(G) = \{X \in \mathbb{C}^{n \times n} : e^{tX} \in G \text{ for all } t \in \mathbb{R}\}.$$

For an abstract Lie group G , $\text{Lie}(G)$ can be defined as the tangent space at the identity element $e \in G$, i.e., $\text{Lie}(G) = T_e G$. This can be endowed with a Lie bracket in a natural way, see [Lee13, Ch. 8].

Definition 2.2.11 (Simple and semisimple). A Lie algebra is called *simple* if it has no non-trivial proper ideals. It is called *semisimple* if it is a direct sum of simple Lie algebras. A connected linear algebraic group $G \subseteq GL(n, \mathbb{C})$ is called *semisimple* if $\text{Lie}(G)$ is semisimple.

Definition 2.2.12 (Tori). Let T be a Lie group. Then T is called a *compact torus* if T is compact, connected, and commutative. For a Lie group K , if a subgroup $T_K \subseteq K$ is maximal with respect to being a compact torus, then we call T_K a maximal compact torus in K .

If T is a commutative linear algebraic group, then T is called an *algebraic torus*. If $G \subseteq GL(n, \mathbb{C})$ is a linear algebraic group and $T \subseteq G$ is a Zariski-closed subgroup which is maximal with respect to being an algebraic torus, then we call T a *maximal algebraic torus* in G .

It can be shown that every compact torus T_K is isomorphic as a Lie group to $U(1)^{\dim_{\mathbb{R}}(T_K)}$, where $U(1) = S^1 = \{z \in \mathbb{C} : |z| = 1\}$ is the circle or unitary group on \mathbb{C}^1 . Similarly, every algebraic torus T is isomorphic as a linear algebraic group to $(\mathbb{C}^\times)^{\dim_{\mathbb{C}}(T)}$. Furthermore, maximal (compact or algebraic) tori have the following properties: they exist, any two of them are conjugate by an element of the containing group, and the Zariski closure of any compact torus in a linear algebraic group G is an algebraic torus [Wal17, Thm. 2.21].

A crucial fact is the following, which gives a characterization of the irreducible representations of tori:

Theorem 2.2.13. Let T be an algebraic torus, $T_K \subseteq T$ a maximal compact torus, and $\varphi: T \rightarrow GL(V)$ a regular representation. Then there exists a unique finite set $\Omega = \Omega(\varphi) \subset \text{Lie}(T)^*$ and a decomposition $V = \bigoplus_{\omega \in \Omega} V_\omega$ into non-empty subspaces V_ω , such that for all $X \in \text{Lie}(T)$ and $v = \sum_{\omega \in \Omega} v_\omega$,

$$\varphi(\exp(X))v = \sum_{\omega \in \Omega} e^{\omega(X)} v_\omega.$$

Here, $\text{Lie}(T)^*$ is the space of \mathbb{C} -linear maps $\text{Lie}(T) \rightarrow \mathbb{C}$, and $\exp: \text{Lie}(T) \rightarrow T$ is the exponential map.

Moreover, if V is endowed with an inner product such that $\varphi(T_K) \subset U(V)$, then the decomposition $V = \bigoplus_{\omega \in \Omega} V_\omega$ is orthogonal.

Definition 2.2.14 (Weights of a representation). The set Ω appearing in Theorem 2.2.13 are referred to as the *weights* of the representation, and the decomposition $V = \bigoplus_{\omega \in \Omega} V_\omega$ as the *weight decomposition*.

More generally, for a connected linear algebraic group $G \subseteq \mathrm{GL}(n, \mathbb{C})$, a choice of maximal algebraic torus $T \subseteq G$, and a regular representation $\pi: G \rightarrow \mathrm{GL}(V)$, we refer to $\Omega(\pi) = \Omega(\pi|_T) \subset \mathrm{Lie}(T)^*$ as the weights of π .

It is often convenient to think of Ω not as a subset of $\mathrm{Lie}(T)^*$ but as a subset of $\mathrm{Lie}(T)$. Assume $T \subseteq \mathrm{GL}(n, \mathbb{C})$ is symmetric, so that $T_K = T \cap \mathrm{U}(n)$. One obtains an identification $\mathrm{Lie}(T) \cong \mathrm{Lie}(T)^*$ via the Hilbert–Schmidt inner product $\langle \cdot, \cdot \rangle$ on $\mathrm{Lie}(T) \subseteq \mathbb{C}^{n \times n}$. Explicitly, if $f \in \mathrm{Lie}(T)^*$, we identify it with $\tilde{f} \in \mathrm{Lie}(T)$ such that

$$\langle \tilde{f}, X \rangle = \mathrm{Tr}[(\tilde{f})^* X] = f(X).$$

for all $X \in \mathrm{Lie}(T)$; recall that our inner products are complex-linear in the *second* argument by convention.

Under this identification, a weight ω becomes an element of $i\mathrm{Lie}(T_K)$: indeed, if $X \in \mathrm{Lie}(T_K)$ and $v_\omega \in V_\omega \setminus \{0\}$, then for all $t \in \mathbb{R}$,

$$\varphi(\exp(tX))v_\omega = e^{t\omega(X)}v_\omega.$$

Therefore if $\omega(X)$ had a non-zero real part, assumed positive without loss of generality, taking the limit as $t \rightarrow \infty$ would cause $\varphi(\exp(tX))v_\omega$ to diverge. However $\varphi(\exp(tX))v_\omega \in \varphi(T_K)v_\omega$ and the latter is a compact set because T_K is compact. As a consequence, $\omega(X) \in i\mathbb{R}$ for $X \in \mathrm{Lie}(T_K) \subseteq i\mathrm{Herm}(n)$, and hence $\mathrm{Tr}[(\tilde{\omega})^* X] \in i\mathbb{R}$ implies that $\tilde{\omega}$ must be Hermitian and in $i\mathrm{Lie}(T_K) \subseteq \mathrm{Herm}(n)$.

Suppose now that $T = (\mathbb{C}^\times)^n \subseteq \mathrm{GL}(n, \mathbb{C})$ is the standard algebraic n -torus and $\varphi: T \rightarrow \mathrm{GL}(V)$ is a representation with weights $\Omega \subset i\mathrm{Lie}(T_K)$. Then the exponential map $\mathrm{Lie}(T) \cong \mathbb{C}^n \rightarrow T$ is given by $\exp(x_1, \dots, x_n) = (e^{x_1}, \dots, e^{x_n})$. This is surjective, and so for $z = (z_1, \dots, z_n) \in T$, there exists $(x_1, \dots, x_n) \in \mathrm{Lie}(T) = \mathbb{C}^n$ such that $z_i = e^{x_i}$, and hence

$$\varphi(z)v_\omega = \varphi(\exp(x))v_\omega = e^{\omega(x)}v_\omega.$$

The right-hand side needs to be independent of the choice of x with $\exp(x) = z$, and in particular for $z = (1, \dots, 1)$, the set of all such x is given by $2\pi i\mathbb{Z}^n$. As a consequence, $\omega(x) \in 2\pi i\mathbb{Z}$ whenever $x \in 2\pi i\mathbb{Z}^n$, and so $\omega \cong \tilde{\omega} \in i\mathrm{Lie}(T_K)$ is naturally in \mathbb{Z}^n (as seen by evaluating $\omega(2\pi i e_j)$ with e_j the standard basis vectors).

For given weights $\Omega \subset \mathbb{Z}^n$, one can explicitly realize a representation V with weights Ω as follows. Let $V \subset \mathbb{C}[u_1, \dots, u_n, u_1^{-1}, \dots, u_n^{-1}]$ be the set of Laurent polynomials which are linear combinations of monomials of the form u^ω with $\omega \in \Omega$. Then $V \cong \mathbb{C}^\Omega$ inherits the inner product from the latter, and admits a T -action by

$$(z_1, \dots, z_n) \cdot u^\omega = z^\omega u^\omega,$$

which clearly has weights given by Ω .

Next, we turn to decompositions of linear algebraic groups. We have the following theorem [Wal17, Thm. 2.22]:

Theorem 2.2.15 (Cartan decomposition). *Let $G \subseteq \mathrm{GL}(n, \mathbb{C})$ be a symmetric linear algebraic group and $K = G \cap \mathrm{U}(n)$. Then for any maximal compact torus $T_K \subseteq K$, its Zariski closure T satisfies $G = KTK$.*

2. Setting the stage

In the special case of $G = \mathrm{GL}(n, \mathbb{C})$ and $K = \mathrm{U}(n)$, this captures the existence of the singular value decomposition. The only difference is that in the singular value decomposition, the element $t \in T$ is only allowed to take positive real values, whereas above they are allowed to be any (non-zero) complex numbers. From this, one can easily deduce the existence of a decomposition as in the following theorem:

Theorem 2.2.16 (Polar decomposition). *Let $G \subseteq \mathrm{GL}(n, \mathbb{C})$ be a symmetric linear algebraic group and $K = G \cap \mathrm{U}(n)$. Then $\mathrm{Lie}(G) = \mathrm{Lie}(K) \oplus i\mathrm{Lie}(K)$, and the map $K \times i\mathrm{Lie}(K) \rightarrow G$ given by $(k, X) \mapsto ke^X$ is a diffeomorphism.*

This is stated in [Wal17, Thm. 2.16] with the weaker assertion that the map is a homeomorphism. The fact that this map is a diffeomorphism (i.e., a smooth bijection with a smooth inverse) is more difficult to prove; we refer to [BH13, Thm. II.10.58].

2.3. Geometry, orbits, and invariants

Geometric invariant theory (GIT) is a field of mathematics that studies orbits of group actions from a perspective that combines geometry and algebra. Inspired by the much earlier work on invariant theory by Hilbert [Hil93], Mumford [MFK94] studied two intimately related questions. The first question is as follows: given an action of a (nice) algebraic group G on an algebraic variety X , how does one construct a *quotient space* X/G as an algebraic object? Although the set of orbits $\{G \cdot x : x \in X\}$ can be endowed with the quotient topology, this is often ill-behaved, and it is not clear how to endow it with a suitable algebraic structure. The second question is how to construct suitable *moduli spaces* of certain algebraic objects (e.g. polarized smooth algebraic curves of a given genus) up to equivalence as algebraic varieties.

We give a gentle introduction to GIT and review some central results (albeit restricting ourselves to a relatively simple setting). These results motivate the algorithmic questions which are of primary interest to us later on. Because of this focus, we also only shortly return to the question of how to construct the previously mentioned quotients (and do not comment on moduli spaces at all).

Throughout this section, we often refer to the textbook on GIT by Wallach [Wal17] and we follow his concrete approach; for a more abstract account see the seminal monograph [MFK94]. We shall work exclusively over \mathbb{C} as a base field.

2.3.1. Mumford's theorem

Let G be a linearly reductive group, let X be an algebraic variety, and let $\sigma: G \times X \rightarrow X$ be an action of G such that σ is a regular map. We shall refer to such an algebraic variety as a G -variety. The idea of a quotient space is that it should be a space whose points are identified with the *orbits* of the action: if $x \in X$, its orbit is given by $G \cdot x = \{\sigma(g, x) : g \in G\} \subseteq X$. While one could formally define the quotient space as just the set of such orbits, it is natural to want to have more structure; in this case, we would like it to be an algebraic variety as well.

We distinguish two different notions of a quotient space. Let $\mathrm{proj}_2: G \times X \rightarrow X$ denote the projection onto the second coordinate.

Definition 2.3.1 (Categorical quotient). A space¹² Y with a morphism $q: X \rightarrow Y$ is called a *categorical quotient* of X by G if:

- (i) $q \circ \sigma = q \circ \text{proj}_2$, i.e., q is G -invariant,
- (ii) for any space Z and morphism $f: X \rightarrow Z$ such that $f \circ \sigma = f \circ \text{proj}_X$, there exists a unique morphism $\hat{f}: Y \rightarrow Z$ such that $f = \hat{f} \circ q$.

A more refined notion of quotient space is as follows:

Definition 2.3.2 (Geometric quotient). A space Y together with a morphism $q: X \rightarrow Y$ is called a *geometric quotient* of X by G if:

- (i) $q \circ \sigma = q \circ \text{proj}_2$,
- (ii) q is a quotient map in the topological sense, i.e., q is surjective and $U \subseteq Y$ is open if and only if $q^{-1}(U)$ is open,
- (iii) for every $y \in Y$, $q^{-1}(y)$ is a *single* G -orbit, and
- (iv) for $U \subseteq Y$ open, $f: U \rightarrow \mathbb{C}$ is a regular function on U if and only if $f \circ q$ is a regular function on $q^{-1}(U)$.

The first important observation to make is that if X is any G -variety and $f: X \rightarrow Z$ is a regular map such that $f \circ \sigma = f \circ \text{proj}_X$, then f is *constant* on orbit closures $\overline{G \cdot x}$.¹³ Therefore, if any orbit $G \cdot x \subseteq X$ is not closed, one has no hope of obtaining a geometric quotient.

Example 2.3.3. Let $X = \mathbb{C}^{m+1} \setminus \{0\}$. Then X is an algebraic variety: it is covered by the affine open sets $U_i = \{(x_0, \dots, x_m) \in \mathbb{C}^{m+1} : x_i \neq 0\}$. The regular functions on U_i are those of the form $(x_0, \dots, x_m) \mapsto q(\frac{x_0}{x_i}, \dots, \frac{x_i}{x_i}, \dots, \frac{x_m}{x_i})$ where $q \in \mathbb{C}[z_1, \dots, z_m]$ is a polynomial. Let \mathbb{C}^\times act on X by

$$\lambda \cdot (x_0, \dots, x_m) = (\lambda x_0, \dots, \lambda x_m).$$

Then m -dimensional projective space \mathbb{P}^m is the geometric quotient of X by this action of \mathbb{C}^\times .

Example 2.3.4. Let \mathbb{C}^\times act on \mathbb{C}^2 by $\lambda \cdot (x_1, x_2) = (\lambda x_1, \lambda^{-1} x_2)$. Then the closed orbits are of the form $\mathbb{C}^\times \cdot (x_1, x_2)$ with $x_1 x_2 \neq 0$, and the orbit $\{0\}$. The other orbits are those of the form $\mathbb{C}^\times \cdot (x_1, x_2)$ with exactly one of x_1 and x_2 non-zero, and 0 is in their orbit closure. The existence of non-closed orbits implies that there is no geometric quotient of \mathbb{C}^2 by \mathbb{C}^\times . However, the categorical quotient is given by \mathbb{C} , with the quotient map $\mathbb{C}^2 \rightarrow \mathbb{C}$ given by $(x_1, x_2) \mapsto x_1 x_2$.

¹²The vague terminology is on purpose. In general, one may need to further enlarge the category of objects for quotients to exist, to e.g. schemes. Moduli spaces in particular may not even exist as schemes, instead falling into the bigger category of algebraic spaces or stacks [Art71; Knu71], see e.g. [Ols16] for a modern introduction.

¹³We take the closure with respect to the Zariski topology here, but note that in the affine or projective setting over \mathbb{C} , the Zariski-closure of $G \cdot x$ agrees with the closure with respect to the Euclidean topology.

2. Setting the stage

Example 2.3.5 (Adjoint action). Let $G = \mathrm{GL}(n, \mathbb{C})$, let $X = \mathbb{C}^{n \times n}$, and let $g \in G$ act on $x \in X$ by $g \cdot x := gxg^{-1}$. Recall that every $x \in X$ can be put into Jordan canonical form: there exists $g \in \mathrm{GL}(n, \mathbb{C})$, $k_1, \dots, k_m \geq 0$ s.t. $\sum_{i=1}^m k_i = n$ and $\lambda_1, \dots, \lambda_m \in \mathbb{C}$ (not necessarily distinct) such that

$$g \cdot x = \bigoplus_{i=1}^m J_{\lambda_i, k_i},$$

where $J_{\lambda, k}$ is the $k \times k$ matrix given by

$$J_{\lambda, k} = \begin{bmatrix} \lambda & 1 & 0 & \dots & 0 \\ 0 & \lambda & 1 & \dots & 0 \\ & & \ddots & \ddots & \\ 0 & 0 & \dots & \lambda & 1 \\ 0 & 0 & \dots & 0 & \lambda \end{bmatrix}.$$

Observe that by acting with diagonal matrices, one can make the off-diagonal entries have arbitrary sizes – in particular, one can send them all to zero. For example, if $a, b \in \mathbb{Z}$, then

$$\begin{bmatrix} 1 & & \\ & z^a & \\ & & z^b \end{bmatrix} \begin{bmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{bmatrix} \begin{bmatrix} 1 & & \\ & z^{-a} & \\ & & z^{-b} \end{bmatrix} = \begin{bmatrix} \lambda & z^{-a} & 0 \\ 0 & \lambda & z^{a-b} \\ 0 & 0 & \lambda \end{bmatrix}.$$

Taking the limit $z \rightarrow \infty$ when $b > a > 0$ shows that $J_{\lambda, 1} \oplus J_{\lambda, 1} \oplus J_{\lambda, 1}$ is in $\overline{G \cdot J_{\lambda, 3}}$. Similarly, $J_{\lambda, 1} \oplus J_{\lambda, 2}$ is also in the orbit closure (using $a = b > 0$).

From the above discussion, it is clear that the image of x under any G -invariant morphism $X \rightarrow Z$ can depend only on the generalized eigenvalues of x . The closed orbits correspond exactly to those $x \in X$ which have a full eigenbasis, i.e., $k_i = 1$ for all $i \in [m]$.

The adjoint action discussed above is particularly nice, because we can exactly determine all the orbits (using the Jordan canonical form). However, one cannot hope to achieve this for general representations. One particularly important setting for which this is still sometimes doable to some extent (but not always) is that of *quiver representations*. We refer to [Rei08] for an overview of this theory in the context of geometric invariant theory.

In this subsection, we establish a fundamental result on orbit closures in the affine setting due to Mumford. Before we state it, we make a definition:

Definition 2.3.6. Let X be an affine G -variety. A (G -)invariant polynomial is a polynomial $p \in \mathbb{C}[X]$ such that, for every $g \in G$ and $x \in X$, $p(g \cdot x) = p(x)$. The invariant ring, denoted by $\mathbb{C}[X]^G$, is the algebra consisting of all G -invariant polynomials.

Theorem 2.3.7 (Mumford). *Let X be an affine G -variety, and let $x, x' \in X$. Then $\overline{G \cdot x}$ contains a unique closed orbit. Furthermore, $\overline{G \cdot x} \cap \overline{G \cdot x'} \neq \emptyset$ (and in particular contain the same unique closed orbit) if and only if $p(x) = p(x')$ for all invariant polynomials $p \in \mathbb{C}[X]^G$.*

It is clear that if $y \in \overline{G \cdot x} \cap \overline{G \cdot x'}$, then $p(x) = p(y) = p(x')$ for every $p \in \mathbb{C}[X]^G$, since p is a continuous G -invariant function and hence constant on orbit closures. The other direction is the interesting one. We start with a fundamental result:

Proposition 2.3.8 ([Wal17, Lem. 3.1]). *Every orbit closure $\overline{G \cdot x}$ contains a closed orbit.*

We explain the idea, and for simplicity assume G is connected. Let $z \in \overline{G \cdot x}$ be such that $\dim(\overline{G \cdot z})$ is minimal. We then claim that $G \cdot z$ is closed, and argue by contradiction. Because G is connected, it is irreducible, hence $G \cdot z$ is irreducible and so is its closure $Y := \overline{G \cdot z}$. Any proper closed subset $Y' \subset Y$ satisfies $\dim(Y') < \dim(Y)$: if $Y_0 \subset \cdots \subset Y_d$ is any chain of distinct closed irreducible subspaces of Y' , then $Y_0 \subset \cdots \subset Y_d \subset Y$ is a chain of distinct closed irreducible subspaces of Y (the Y_i are closed in Y because Y' is). Now $G \cdot z$ is automatically Zariski-open in $\overline{G \cdot z}$, as a consequence of Chevalley's theorem on constructible sets, see [Wal17, Lem. 3.1]. Therefore if $Y \neq G \cdot z$, then $Y' = Y \setminus (G \cdot z)$ is closed and non-empty. Then any $y \in Y'$ satisfies $\dim(\overline{G \cdot y}) \leq \dim(Y')$ (here we use that for arbitrary $Z \subset Y'$, one has $\dim(Z) \leq \dim(Y')$). Since $\dim(Y')$ is strictly smaller than $\dim(Y) = \dim(\overline{G \cdot z})$, we see that the latter is not minimal.

Next, we recall that any given a variety, any two disjoint affine subvarieties can be separated by regular function:

Lemma 2.3.9 ([Wal17, Thm. 3.12]). *Let X be an affine variety and let $Y, Z \subseteq X$ be disjoint affine subvarieties. Then there exists $p \in \mathbb{C}[X]$ such that $p|_Y \equiv 0$ and $p|_Z \equiv 1$.*

Proof. Let $J \subseteq \mathbb{C}[X]$ denote the ideal of polynomials vanishing on Y . Then $L = J|_Z$ is an ideal in $\mathbb{C}[Z]$.¹⁴ Because $Y \cap Z = \emptyset$, for every $z \in Z$ there must exist $p \in L$ such that $p(z) \neq 0$, as otherwise we would have $z \in Y$. Therefore we must have $L = \mathbb{C}[Z]$ by the Nullstellensatz (Theorem 2.2.1), as any proper ideal in $\mathbb{C}[Z]$ would have non-empty common zero set. In particular, there exists some $p \in J$ such that $p|_Z = 1 \in \mathbb{C}[Z]$. \square

The goal now is to upgrade Lemma 2.3.9 to a setting where all objects are G -invariant, in the following sense: we replace X, Y, Z by affine varieties upon which G acts, and we replace arbitrary polynomial functions on X by *invariant polynomials* on X . The linear reductivity of G provides us with the following ingredient:¹⁵

Proposition 2.3.10. *Let G be a linearly reductive group. Then for every affine G -variety X , there exists an operator $R_X: \mathbb{C}[X] \rightarrow \mathbb{C}[X]^G$ called the Reynolds operator, with the following properties:*

- (i) $R_X(1) = 1$.
- (ii) If $p \in \mathbb{C}[X]$ and $q \in \mathbb{C}[X]^G$, then $R_X(qp) = qR_X(p)$.
- (iii) If $Y \subseteq X$ is a G -invariant Zariski-closed subset, then for every $p \in \mathbb{C}[X]$, $R_Y(p|_Y) = R_X(p)|_Y$.

We sketch the proof. The idea is to think of $\mathbb{C}[X]$ as a regular representation of G , under the action $(g \cdot p)(x) := p(g^{-1} \cdot x)$. Then $\mathbb{C}[X]^G$ is the maximal subspace on which G acts trivially, i.e., the maximal trivial subrepresentation, and so it

¹⁴We use here that the varieties are affine: the ring homomorphism $\mathbb{C}[X] \rightarrow \mathbb{C}[Z]$ given by restriction is surjective (by definition), hence the *image* of the $\mathbb{C}[X]$ -ideal $J \subseteq \mathbb{C}[X]$ under this ring homomorphism is also a $\mathbb{C}[Z]$ -ideal in $\mathbb{C}[Z]$.

¹⁵We restrict the statement to coordinate rings $\mathbb{C}[X]$ for convenience, but such an operator also exists for general “dual actions”, see [MFK94, Sec. 1.1].

2. Setting the stage

would be enough to construct a projection onto this subspace. One complication is that $\mathbb{C}[X]$ is usually infinite-dimensional, so it is not obvious that there is a well-defined (or canonical) complementary subspace, and hence no obvious way to construct a projection $R_X: \mathbb{C}[X] \rightarrow \mathbb{C}[X]^G$. However, one has the following useful ingredients:

- (i) If $V \subseteq \mathbb{C}[X]$ is a *finite-dimensional* subspace, then there exists a finite-dimensional G -invariant subspace $V' \subseteq \mathbb{C}[X]$ containing V .
- (ii) Every finite-dimensional regular representation V of G splits as a direct sum $V = V^G \oplus W$, where V^G consists of those $v \in V$ such that $g \cdot v = v$ for every $g \in G$, and W is the sum of all irreducible non-trivial subspaces of V [Wal17, Lem. 3.6].

Then one can define the Reynolds operator R_X as follows. For $p \in \mathbb{C}[X]$, let $V \subseteq \mathbb{C}[X]$ be a finite-dimensional invariant subspace containing p . Then let $R_X(p)$ be the image of p under the projection $V = V^G \oplus W \rightarrow V^G$. Verifying that R_X is a linear operator with the desired properties is left as an exercise to the reader.

The above description of R_X is rather abstract. For $GL(n, \mathbb{C})$ or $SL(n, \mathbb{C})$, more explicit descriptions exist via Cayley's Ω -process; see for instance [Stu08; DK15]. This explicit description is also useful for analyzing algorithms for scaling problems, see [BFG+19].

From Lemma 2.3.9 and the existence of the Reynolds operator R_X as in Proposition 2.3.10, one deduces the following separation theorem:

Theorem 2.3.11 (Mumford, [MFK94, Cor. 1.2]). *Let G be a linearly reductive group, let X be an affine G -variety and let $Y, Z \subseteq X$ be G -invariant disjoint affine subvarieties. Then there exists $p \in \mathbb{C}[X]^G$ such that $p|_Y \equiv 0$ and $p|_Z \equiv 1$.*

Together with Proposition 2.3.8, this proves Theorem 2.3.7: if $G \cdot y \subseteq \overline{G \cdot x}$ and $G \cdot y' \subseteq \overline{G \cdot x'}$ are closed orbits, then $p(x) = p(y)$ and $p(x') = p(y')$ for every $p \in \mathbb{C}[X]^G$. Therefore if $p(x) = p(x')$ for every $p \in \mathbb{C}[X]^G$, we have $p(y) = p(y')$, so $G \cdot y$ and $G \cdot y'$ cannot be separated by a G -invariant regular function; by Theorem 2.3.11, $G \cdot y$ and $G \cdot y'$ cannot be disjoint, and in fact must be equal (since two group orbits intersect if and only if they are equal). Therefore $G \cdot y \subseteq \overline{G \cdot x} \cap \overline{G \cdot x'}$.

2.3.2. The invariant ring

We next discuss a classical fact about invariant rings due to Hilbert [Hil93].

Theorem 2.3.12 (Finite generation). *Let X be an affine G -variety. Then the invariant ring $\mathbb{C}[X]^G$ is a finitely generated algebra.*

This can be proven using the existence of Reynolds operators, see [Wal17, Thm. 3.11]. In short, because $\mathbb{C}[X]$ is Noetherian, the ideal $J \subseteq \mathbb{C}[X]$ generated by $\mathbb{C}[X]^G$ admits finitely many generators in $\mathbb{C}[X]^G$. The Reynolds operators can then be used to show that these also generate $\mathbb{C}[X]^G$ as an algebra.

Since $\mathbb{C}[X]^G$ is a subalgebra of $\mathbb{C}[X]$, it also has trivial nilradical. Therefore, the above theorem shows that $\mathbb{C}[X]^G$ is the coordinate ring of an affine variety! In fact, the affine variety defined by it is a categorical quotient of V by G [MFK94,

Thm. 1.1], thus proving that *affine G-varieties have affine categorical quotients*. This quotient is a geometric quotient if and only if every orbit $G \cdot x$ is closed [MFK94, Amp. 1.3, Def. 0.8] (which is clearly a necessary condition, by the discussion below Definition 2.3.2).

We give some examples of invariant rings:

Example 2.3.13 (Torus action). Let $X = \mathbb{C}^k$, let $\omega_1, \dots, \omega_k \in \mathbb{Z}^n$ and let $G = (\mathbb{C}^\times)^n$ act on X via

$$(z_1, \dots, z_n) \cdot (x_1, \dots, x_k) = (z^{\omega_j} x_j)_{j=1}^k, \quad z^{\omega_j} := z_1^{(\omega_j)_1} \cdots z_n^{(\omega_j)_n}.$$

Note that the action of G is linear on X , hence is a representation. The vectors $\omega_1, \dots, \omega_k$ are exactly the weights associated with the representation (see Definition 2.2.14).

The invariant ring $\mathbb{C}[X]^G$ consists of those $p \in \mathbb{C}[X]$ which are linear combinations of monomials $x^a = x_1^{a_1} \cdots x_k^{a_k}$, $a \in \mathbb{Z}_{\geq 0}^k$, such that $\sum_{i=1}^k a_i \omega_i = 0$. Therefore a generating set of invariants can be obtained from a *Hilbert basis* for the set $A = \{a \in \mathbb{Z}_{\geq 0}^k : \sum_{i=1}^k a_i \omega_i = 0\}$, i.e., a (necessarily unique) minimal set of a^1, \dots, a_r such that every $a \in A$ is a non-negative integer combination of the a^i . Writing down such a Hilbert basis is computationally hard in general, see [Stu08, Sec. 1.4] for more details and Chapter 3 for an explicit example.

Example 2.3.14 (Matrix scaling). Let $G = \text{ST}(n) \times \text{ST}(n)$ where $\text{ST}(n) \subseteq \text{GL}(n, \mathbb{C})$ denotes the group of $n \times n$ diagonal matrices with unit determinant. We let G act on $V = \mathbb{C}^{n \times n}$ by

$$(X, Y) \cdot A = XAY.$$

The Lie algebra $\text{Lie}(G)$ is given by pairs (U, V) with $U, V \in \mathbb{C}^{n \times n}$ diagonal matrices with $\text{Tr}[U] = \text{Tr}[V] = 0$, and the exponential map $\exp: \text{Lie}(G) \rightarrow G$ is given by $(U, V) \mapsto (e^U, e^V)$ (interpreted as matrix exponentiation, but this agrees with exponentiating the diagonal entries of U). Note that G is an algebraic torus of dimension $2(n-1)$. Because

$$(e^U, e^V) \cdot A = e^U A e^V = \sum_{i,j=1}^n e^{U_{ii} + V_{jj}} A_{ij} e_i e_j^T,$$

the weights of the representation are given by the forms $\omega_{ij} \in \text{Lie}(T)^*$ defined by

$$\omega_{ij}(U, V) = U_{ii} + V_{jj}.$$

Identifying $\text{Lie}(T)^* \cong \text{Lie}(T)$ yields

$$\omega_{ij} = (e_i e_i^T - \frac{1}{n} I, e_j e_j^T - \frac{1}{n} I);$$

note that the $-\frac{1}{n}$ appears because $\text{Lie}(T)$ consists of pairs of *traceless* matrices.

Now assume that $p \in \mathbb{C}[V]$ is an invariant polynomial; then as in Example 2.3.13, p is a sum of invariant *monomials* in the entries of A , i.e., expressions of the form

$$\prod_{\ell=1}^k A_{i_\ell j_\ell}$$

2. Setting the stage

for pairs $(i_1, j_1), \dots, (i_k, j_k) \in [n]^2$. A monomial is invariant under the action if and only if $\prod_{\ell=1}^k X_{i_\ell} Y_{j_\ell}$ is constant over all $(X, Y) \in G$; but that implies that this product is a power of $\det(X) \det(Y)$, i.e., every $i \in [n]$ and $j \in [n]$ appears the same number of times among the i_1, \dots, i_k and j_1, \dots, j_k respectively. As a result, a generating set for the invariants of the action is given by the *matching monomials* $A \mapsto \prod_{\ell=1}^n A_{\ell\sigma(\ell)}$, where $\sigma \in S_n$ is a permutation.

We note here that the sum of the squared absolute values of this generating set is the *permanent* of the matrix with (i, j) -th entry given by $|A_{ij}|^2$, which made an appearance as a progress measure for analyzing Sinkhorn's algorithm for matrix scaling in [LSW00]. From Mumford's theorem (Theorem 2.3.7) one deduces that this permanent is positive if and only if $0 \in \overline{G \cdot A}$, which is true if and only if the support of A contains a bipartite perfect matching.

Example 2.3.15 (Adjoint action). Let $X = \mathbb{C}^{n \times n}$ and let $G = \text{GL}(n, \mathbb{C})$ act on X by $g \cdot x := gxg^{-1}$. Then for every $k \geq 0$, the polynomial $x \mapsto \text{Tr}[x^k]$ is a G -invariant polynomial by cyclicity of the trace. In fact, essentially all of the invariant polynomials are of this form: $\mathbb{C}[X]^G$ is generated (as an algebra) by these maps. This is due to Weyl [Wey46], see [Pro76] for a more accesible proof using Schur–Weyl duality. We shall study this example and generalizations of it in more detail in Chapter 3.

One can use the Cayley–Hamilton theorem to bound the number of generators needed: the characteristic polynomial $\det(\lambda I - A)$ is of the form $p_0(A) + \dots + \lambda^n p_n(A)$ for some *invariant* polynomials $p_0, \dots, p_n \in \mathbb{C}[x_{11}, x_{12}, \dots, x_{nn}]$ (since the characteristic polynomial is invariant under conjugation). These polynomials are exactly the elementary symmetric polynomials in the (generalized) eigenvalues of A : if $\lambda_1, \dots, \lambda_n$ are the eigenvalues, then

$$p_i(A) = e_i(\lambda_1, \dots, \lambda_n)$$

where e_i is the i -th elementary symmetric polynomial in n variables. Now observe that the traces $\text{Tr}[A^k]$ are *power sums* of the eigenvalues, since $\text{Tr}[A^k] = \lambda_1^k + \dots + \lambda_n^k$. It is well-known that both the elementary symmetric polynomials e_i for $1 \leq i \leq n$, and the power sum polynomials of degree $1 \leq i \leq n$, form a generating set for the ring of symmetric polynomials in n variables, as can be deduced from e.g. [Sag13, Thm. 4.3.7]. Moreover, for $k \geq n$,

$$A^k = A^{k-n} A^n = A^{k-n} \left(-p_0(A) - \dots - p_{n-1}(A) A^{n-1} \right)$$

by the Cayley–Hamilton theorem, and hence $\text{Tr}[A^k]$ can be expressed as a polynomial in $\text{Tr}[A^j]$ for $j < k$ by induction.

The fact that the invariants here are all given by symmetric polynomials in the eigenvalues is a phenomenon that occurs more generally. Let G be a connected semisimple algebraic Lie group, $\text{Lie}(G)$ its Lie algebra endowed with the adjoint action of G , T a maximal algebraic torus in G , $\text{Lie}(T)$ the corresponding Cartan subalgebra, and $W = N(T)/T$ the Weyl group where $N(T)$ is the normalizer of T in G . Then *Chevalley's restriction theorem* (see e.g. [Wal17, Thm. 3.62]) states that the natural algebra morphism $\mathbb{C}[\text{Lie}(G)]^G \rightarrow \mathbb{C}[\text{Lie}(T)]^W$ induced by restriction is an isomorphism. For $G = \text{SL}(n, \mathbb{C})$, $\text{Lie}(G)$ consists of the traceless $n \times n$ matrices, T are the diagonal $n \times n$ matrices with determinant 1 and $\text{Lie}(T)$ the traceless

diagonal $n \times n$ matrices, and $W \cong S_n$ acts on $\text{Lie}(T)$ by permuting the diagonal elements. Therefore $\mathbb{C}[\text{Lie}(T)]^W$ consists of the symmetric polynomials in the diagonal entries.

2.3.3. The Hilbert–Mumford theorem

In the proof of Proposition 2.3.8, we saw that the (unique) closed orbit $G \cdot w$ in an orbit closure $\overline{G \cdot v}$ is characterized as the unique orbit (closure) of minimal dimension in $\overline{G \cdot v}$. Let $G_w = \{g \in G : g \cdot w = w\}$ be the stabilizer of w . Then G_w is also a linear algebraic group (the equation $g \cdot w = w$ defines a Zariski-closed subset of G) and hence has a well-defined dimension; by orbit-stabilizer reasoning, one expects that $\dim(G) = \dim(G \cdot w) + \dim(G_w)$. Therefore if the dimension of $G \cdot w$ is minimal, $\dim(G_w)$ is maximal, and in particular larger than $\dim(G_v)$. It turns out that this can in fact be “witnessed” by a 1-parameter subgroup of G , in the following sense:

Theorem 2.3.16 (Extended Hilbert–Mumford criterion). *Let $\pi: G \rightarrow \text{GL}(V)$ be a regular representation, let $v \in V$ and let $G \cdot w \subseteq \overline{G \cdot v}$ be the unique closed orbit. Then there exists an algebraic group homomorphism $\varphi: \mathbb{C}^\times \rightarrow G$ such that $\lim_{z \rightarrow 0} \varphi(z) \cdot v \in G \cdot w$. Furthermore, if G is chosen to be a symmetric subgroup of $\text{GL}(n, \mathbb{C})$, then φ can be chosen such that $\varphi(\bar{z}) = \varphi(z)^*$.*

The theorem in the case where $w = 0$ (i.e., $0 \in \overline{G \cdot v}$) is what is usually known as the Hilbert–Mumford criterion, and is due to Hilbert [Hil93, p. V.18] in the case of $G = \text{GL}(n, \mathbb{C})$, and Mumford for general G [MFK94, Thm. 2.1]. This general form also appears implicitly in the proof [MFK94, p. 53], explicitly in Kempf [Kem78, Thm. 1.4], and was also proven by Richardson [Bir71, Thm. 4.2] using different methods. Birkes [Bir71] showed that a version of the criterion also holds over \mathbb{R} . For an accessible proof we refer to [Wal17, Thm. 3.24]

The above result serves as an incredible computational criterion, for the following reason: suppose one wants to characterize all $v \in V$ such that $0 \in \overline{G \cdot v}$; we refer to such v as being *G-unstable* (we explain why this notion is interesting in the next subsection). To achieve this, first fix a maximal algebraic torus $T \subseteq G$. Then it is often feasible to explicitly determine those $w \in V$ with $0 \in \overline{T \cdot w}$. Now let $v \in V$ be such that $0 \in \overline{G \cdot v}$. Then the Hilbert–Mumford criterion yields some $\varphi: \mathbb{C}^\times \rightarrow G$ with $\lim_{z \rightarrow 0} \varphi(z) \cdot v = 0$. Since any two maximal algebraic tori are conjugate [Wal17, Thm. 2.21], there exists some $g \in G$ such that $g\varphi(z)g^{-1} \in T$ for all $z \in \mathbb{C}^\times$; but this implies that $g \cdot v$ is *T-unstable*! Therefore the union of the G -orbits of *T-unstable* vectors is exactly the set of *G-unstable* vectors, and one can hope to give a “basis-invariant” characterization of *G-unstable* vectors by appropriately rephrasing *T-instability*. We give some examples in Section 2.3.4.

To see how the theorem connects to discussion at the start of this subsection, assume that $\overline{G \cdot v}$ is not closed, and that $\dim(G_v) = 0$. If $w' := \lim_{z \rightarrow 0} \varphi(z) \cdot v$, then $\varphi(\mathbb{C}^\times) \subseteq G_{w'}$. Therefore $\dim(G_{w'}) \geq 1 > 0 = \dim(G_v)$. Since $w' = g \cdot w$ for some $g \in G$, $G_{w'} = gG_wg^{-1}$ is isomorphic to G_w , and hence $\dim(G_w) \geq 1$.¹⁶

¹⁶It is unclear how to generalize this argument to $\dim(G_v) \geq 1$; even though $\varphi(\mathbb{C}^\times)$ is not in G_v , it could still be the case that $G_{\varphi(z)v} = \varphi(z)G_v\varphi(z)^{-1}$ “converges” to a subgroup of $G_{w'}$, and hence $\varphi(\mathbb{C}^\times)$ would not be an actual witness to the growth in dimension. However, when $\varphi(\mathbb{C}^\times)$

2.3.4. Stability and quotients of projective varieties

It is very often natural to consider projective varieties $X \subseteq \mathbb{P}^m$ rather than affine varieties, as they are better behaved in many ways; for instance, they are complete, which is the algebraic analog of compactness. Taking quotients in this context is a more delicate procedure, however. One of the reasons is that unlike for the affine setting, any globally defined regular function is constant, so $\mathbb{C}[X] = \mathbb{C}$ and we have no useful direct analog of $\mathbb{C}[X]^G$.

Let us be somewhat more precise. Consider a regular representation $\pi: G \rightarrow GL(V)$ of a symmetric linear algebraic group $G \subseteq GL(n, \mathbb{C})$. Then this descends to an action of G on $\mathbb{P}(V)$. Let $X \subseteq \mathbb{P}(V)$ be a G -invariant closed subset of $\mathbb{P}(V)$. Then we can attempt to construct a quotient of X as follows. The set $X' \subseteq V$ defined by

$$X' = \{v \in V : [v] \in X \text{ or } v = 0\}$$

is an affine variety in V : if $p_1, \dots, p_r \in \mathbb{C}[V]$ are homogeneous polynomials whose common zero set on $\mathbb{P}(V)$ is X , then X' is their common zero set in V . It is easy to verify that X' is also G -invariant. Then we define the *GIT quotient* $X//G$ as the projectivization $\text{Proj } R$ of the graded ring $R = \bigoplus_{d \geq 0} \mathbb{C}[X']_d^G$, where $\mathbb{C}[X']_d^G$ is the set of G -invariant homogeneous polynomials of degree d . The points of $\text{Proj } R$ are given by homogeneous ideals in R which maximal with respect to the property of not containing the *irrelevant ideal* $\bigoplus_{d > 0} \mathbb{C}[X']_d^G$, see [Har77] for details. This corresponds to the fact that a projective space $\mathbb{P}(V)$ is formed by *first removing the origin*, and then identifying points on the same line. Note also that $\mathbb{C}[X']_0^G = \mathbb{C}[X']_0$ consists of the constant functions on X' , hence is isomorphic to \mathbb{C} . It can be turned into an algebraic variety in a similar manner as for projective space $\mathbb{P}(V)$, which is the projectivization $\text{Proj } \mathbb{C}[V]$.

In the G -invariant setting, we must therefore *first remove the points equivalent to zero*, under the relation given by $v \sim w$ if and only if $\overline{G \cdot v} \cap \overline{G \cdot w} \neq \emptyset$. This motivates (part of) the following terminology:

Definition 2.3.17 (Stability). Let $\pi: G \rightarrow GL(V)$ be a regular representation. A vector $v \in V \setminus \{0\}$ is called¹⁷

- (i) *unstable* if $0 \in \overline{G \cdot v}$,
- (ii) *semistable* if $0 \notin \overline{G \cdot v}$,
- (iii) *polystable* if $G \cdot v$ is closed, and
- (iv) *stable* if v is polystable and the stabilizer G_v of v is finite.

The *semistable locus* X^{ss} , the *polystable locus* X^{ps} and the *stable locus* $X^s \subseteq X$ consist of those $x = [v] \in X$ such that v is semistable, polystable or stable, respectively. The set of $v \in V$ with v unstable or 0 is referred to as the *null-cone* of the action on V .

is in the normalizer of G_v , this is the case; in particular, this interpretation is valid when G is commutative (i.e. $G = T$). Furthermore, many actions encountered in the wild are such that generic points have finite stabilizer.

¹⁷In [MFK94], *properly stable* is used for what we call stable, and *stable* is used for v such that there is an open neighbourhood $U \subseteq X^{ss}$ of $[v]$ containing only polystable points.

The usefulness of unstable versus semistable is clear from the previous discussion. To appreciate the value of a vector being stable, observe that by the extended Hilbert–Mumford criterion (Theorem 2.3.16), if $v, w \in V \setminus \{0\}$ are vectors such that $v \in \overline{G \cdot w} \setminus (G \cdot w)$ and $G \cdot v$ is closed, the stabilizer G_v of v contains a 1-parameter subgroup of G (see Section 2.3.3). Therefore G_v will not be finite. From the perspective of taking quotients, this means that the orbit of a vector which is stable is guaranteed not to be identified with any other orbit. Moreover, the polystable orbits are in one-to-one correspondence with the points in the set-theoretic quotient X^{ss}/G by Theorem 2.3.7.

Generally, the GIT quotient $q: X \rightarrow X//G = X^{ss}/G$ is a categorical quotient. If there exist stable points in the sense of Mumford [MFK94, Def. 1.7], then there is a non-empty open subset $U \subseteq X^{ss}/G$ such $q^{-1}(U)$ consists of all Mumford-stable points, and $q|_{q^{-1}(U)}: q^{-1}(U) \rightarrow U$ is a geometric quotient [MFK94, Thm. 1.10]. The set $q^{-1}(U)$ contains all stable points.

We note here that [MFK94] constructs this quotient in a more general setting: rather than assuming X to be projective, one can take an arbitrary scheme, together with an *invertible sheaf* L on X such that G also linearly acts on L in a way that lifts the action on X . This data is referred to as a G -linearization. When $X \subseteq \mathbb{P}(V)$ is projective and the action of G comes from a linear action on V , such a G -linearization of the action can be obtained as follows: invertible sheaves are the algebraic analogue of line bundles, and over $\mathbb{P}(V)$ one has a *tautological bundle* $\tau \rightarrow \mathbb{P}(V)$ whose fiber over $[v] \in \mathbb{P}(V)$ is given by $\mathbb{C} \cdot v \subseteq V$ (the corresponding invertible sheaf is often denoted by $\mathcal{O}(-1)$). The *global sections* of the *dual* L of the tautological line bundle (corresponding to $\mathcal{O}(1)$) are linear functions on V , i.e., $\mathbb{C}[V]_1$. More generally, global sections of $L^{\otimes d}$ are homogeneous degree d polynomials. One can define the various notions of stability relative to a G -linearization L by using G -invariant sections of $L^{\otimes d}$, see [MFK94, Def. 1.7]; that this generalizes Definition 2.3.17 then follows from Theorem 2.3.7.

Example 2.3.18 (Adjoint action). Let $G = \mathrm{SL}(n, \mathbb{C})$ act on $V = \{x \in \mathbb{C}^{n \times n} : \mathrm{Tr}[x] = 0\}$ by $g \cdot x = gxg^{-1}$, and let $X = \mathbb{P}(V)$. Then any matrix in $G \cdot x$ has the same eigenvalues as x , and so if x is unstable, its eigenvalues must all be zero, i.e., x is nilpotent. Clearly any nilpotent x is also unstable: first put x into Jordan canonical form, then one can push the off-diagonal elements to zero as in Example 2.3.5. In other words, the unstable vectors are exactly the nilpotent matrices. The polystable vectors are those with all Jordan blocks of size one, i.e., having a full eigenbasis. There are no stable vectors: if x has a full eigenbasis, then any matrix $g \in \mathrm{SL}(n, \mathbb{C})$ which is diagonal in this basis commutes with x , so $g \cdot x = gxg^{-1} = x$. In particular, the stabilizer G_x contains an algebraic torus of dimension $n - 1$, so is not finite unless $n = 1$. It is however true that matrices x with n distinct eigenvalues are not in $\overline{G \cdot x'} \setminus (G \cdot x')$ for any x' , as can be deduced from the Jordan canonical form and the Hilbert–Mumford criterion Theorem 2.3.16, see Example 2.3.5.

The set of x with n distinct eigenvalues form the non-vanishing set of the *discriminant* of the characteristic polynomial of x , which is G -invariant (because the characteristic polynomial is). This set is also exactly the set of stable vectors in the sense of Mumford.

Example 2.3.19 (Binary forms). We consider the classical example of binary forms of degree d [MFK94, Sec. 4.1]. Let $V = \mathbb{C}[x, y]_d = \mathrm{Sym}^d((\mathbb{C}^2)^*)$ denote the space

2. Setting the stage

of homogeneous degree d polynomials in two variables x, y . Then $G = \mathrm{SL}(2, \mathbb{C})$ acts on V by $(g \cdot p)(x, y) = p(g^{-1}(x, y))$, where $g^{-1} \cdot (x, y)$ is defined by matrix multiplication on \mathbb{C}^2 . Let $T = \{\mathrm{diag}(z, z^{-1}) : z \in \mathbb{C}^\times\}$ be the maximal algebraic torus in G . Note that every 1-parameter subgroup of G is conjugate to a subgroup of T , so by the Hilbert–Mumford criterion (Theorem 2.3.16), the set of unstable vectors $p \in V$ is exactly given by the G -orbits of the T -unstable vectors.

We now determine the $p \in V$ which are T -unstable. Suppose $p(x, y) = \sum_{i=0}^d a_i x^i y^{d-i}$ for $a_i \in \mathbb{C}$. Then p is T -unstable if and only if either $a_i = 0$ for all $i \leq d/2$ or $a_i = 0$ for all $i \geq d/2$. To see this, suppose that for all $x, y \in \mathbb{C}$,

$$\lim_{z \rightarrow 0} ((\mathrm{diag}(z^{-1}, z) \cdot p)(x, y)) = \lim_{z \rightarrow 0} \sum_{i=0}^d a_i z^{d-2i} x^i y^{d-i} = 0.$$

Then $a_i z^{d-2i} \rightarrow 0$ as $z \rightarrow 0$ for all $i = 0, \dots, d$, and in particular $a_i = 0$ whenever $d - 2i \leq 0$. Similarly, the above limit being zero as $z \rightarrow \infty$ would imply $a_i = 0$ whenever $d - 2i \geq 0$.

To characterize their G -orbits, we observe the following: every $p \in V \setminus \{0\}$ admits a (not necessarily unique) factorization

$$p(x, y) = \prod_{j=0}^d (b_j x + c_j y)$$

as a product of linear forms, for some coefficients $b_j, c_j \in \mathbb{C}$. It is enough to prove this by induction on d , and the statement is trivial for $d = 1$. For higher d , observe that for $\lambda \in \mathbb{C}$,

$$\left(\begin{bmatrix} 1 & 0 \\ \lambda & 1 \end{bmatrix} \cdot p \right)(x, y) = p(x, y - \lambda x) = \sum_{i=0}^d a_i x^i (y - \lambda x)^{d-i},$$

whose coefficient of x^d is some polynomial in λ . By the fundamental theorem of algebra, there exists some $\lambda_0 \in \mathbb{C}$ for which $p(x, y - \lambda_0 x)$ has vanishing coefficient for x^d (except in the case where the coefficient is constant as a function of λ , but then p is a multiple of x^d), and hence factorizes as $p(x, y - \lambda_0 x) = y q(x, y)$ for some $q \in \mathbb{C}[x, y]_{d-1}$. But then $p(x, y) = (y + \lambda_0 x) q(x, y + \lambda_0 x)$ is a factorization of p .

From this factorization, one can deduce that p is G -unstable if and only if some linear factor appears strictly more than $d/2$ in the factorization, where two linear factors are considered equivalent if and only if differ by a scalar multiple. The semistable points are those where every linear factor occurs at most $d/2$ times, and the polystable points are those where this inequality is strict.

The GIT quotient $\mathbb{P}(V)//G$ can be interpreted as the “moduli space of d points on \mathbb{P}^1 ”, as every linear factor specifies a unique point in \mathbb{P}^1 . The existence of the factorization can also be deduced abstractly by observing that the map $(\mathbb{P}^1)^d \rightarrow \mathbb{P}^d = \mathbb{P}(V)$ given by multiplying the relevant linear forms is smooth, has irreducible image and has derivative of rank d at a generic point, so the image must be all of \mathbb{P}^d . This map is also $\mathrm{SL}(2, \mathbb{C})$ -equivariant, and we have used this above to give a “constructive” proof of the existence of such a factorization.

2.4. The Kempf–Ness theorem

We now return to the problem of classifying orbit closures. Let G be a connected symmetric linear algebraic group, $\pi: G \rightarrow GL(V)$ a regular representation, K a maximal compact subgroup of G , and assume V is endowed with a K -invariant inner product. We shall denote the induced norm on V by $\|\cdot\|$. Recall that Theorem 2.3.7 shows that two orbit closures intersect if and only if they contain the same closed orbit; in a sense, the closed orbit classifies the orbit closure. However, it is still desirable to understand whether one can give an essentially *canonical* representative *within* the closed orbit. The Kempf–Ness theorem [KN79] achieves this. For convenience, we use the following terminology, which is not entirely standard but is natural:¹⁸

Definition 2.4.1 (Minimum norm vectors). For $v \in V$, we say that v_{\min} is a *minimum norm vector* for v if

$$v_{\min} \in \operatorname{argmin}\{\|w\| : w \in \overline{G \cdot v}\}.$$

That is, v_{\min} is a minimum norm vector for v if $v_{\min} \in \overline{G \cdot v}$ and $\|v_{\min}\| = \min_{w \in \overline{G \cdot v}} \|w\| = \inf_{g \in G} \|g \cdot v\|$.

Clearly, any vector $v \in V$ has a minimum norm vector v_{\min} : the set $\{w \in \overline{G \cdot v} : \|w\| \leq \|v\|\}$ is compact (closed and bounded) with respect to the Euclidean topology, and $\|\cdot\|$ is a continuous function thereon, so achieves its minimum. The minimum norm vectors are in general *not* unique, since if v_{\min} is a minimum norm vector then so is $k \cdot v_{\min}$ for any $k \in K$ (recall that K preserves the inner product, hence also the norm). Crucially, this is the *only* source of non-uniqueness, as we shall see shortly.

It is also convenient to make the following definition:

Definition 2.4.2. Let $v \in V \setminus \{0\}$. Then the *Kempf–Ness function* $F_v: G \rightarrow \mathbb{R}$ associated with v is given by

$$F_v(g) = \log\|g \cdot v\|.$$

Note that taking the logarithm is well-defined since G acts by invertible linear transformations, hence $g \cdot v$ is never zero. Observe that if v_{\min} is a minimum norm vector for v , then $v_{\min} = 0$ and F_v is unbounded from below, or $F_v(g) \geq \log\|v_{\min}\|$ for all $g \in G$.

We now focus on the properties of the minimum norm vector itself. It is clear that if w is a vector of minimal norm in an orbit closure, then it is in particular a vector of minimal norm in its own G -orbit, hence the derivatives of the norm (or norm squared) must vanish in any direction along the orbit. These directions are given by the Lie algebra $\operatorname{Lie}(G)$ of G , which is the complex vector space consisting of all matrices $X \in \mathbb{C}^{n \times n}$ such that $e^{tX} \in G$ for all $t \in \mathbb{R}$ (see Definition 2.2.10). Then $t \mapsto e^{tX} \cdot w$ is a smooth curve in the orbit of w . Accordingly, if w is a vector of minimum norm in its orbit, then $\|e^{tX} \cdot w\|^2$ must have a minimum at $t = 0$ and the derivative at $t = 0$ will vanish. This motivates the following definition:

Definition 2.4.3. A vector $w \in V$ is called *critical* if $\partial_{t=0}\|e^{tX} \cdot w\|^2 = 0$ for every $X \in \operatorname{Lie}(G)$, or equivalently if $w \neq 0$, $\partial_{t=0}F_w(e^{tX}) = 0$ for every $X \in \operatorname{Lie}(G)$.

¹⁸For instance, [NM84] uses the term “minimal vector”.

2. Setting the stage

Since K acts unitarily, the norm will always be preserved if we move in directions that keep us in the K -orbit; these are given by the Lie algebra $\text{Lie}(K)$ of K . As $K = G \cap U(n) \subseteq GL(n, \mathbb{C})$, $\text{Lie}(K)$ consists of those $X \in \text{Lie}(G)$ such that X is skew-Hermitian. Furthermore, $\text{Lie}(G)$ splits as a direct sum $\text{Lie}(G) = \text{Lie}(K) \oplus i\text{Lie}(K)$ (as a vector space, but not as a Lie algebra). One can show that Definition 2.4.3 is equivalent to demanding that $\partial_{t=0} \|e^{tX} \cdot w\|^2 = 0$ for all $X \in i\text{Lie}(K)$; the latter are precisely the Hermitian matrices in $\text{Lie}(G)$.

Criticality is the natural first-order condition for a vector to have minimum norm in its orbit (“at a minimum, all derivatives vanish”). Remarkably, this is also *sufficient!* This was shown by Kempf and Ness [KN79], which further characterizes the existence of minimum norm vectors. The precise statement is as follows, see also [Wal17, Thm. 3.26]:

Theorem 2.4.4 (Kempf–Ness). *Let $v \in V$. Then:*

- (i) v is critical if and only if $\|g \cdot v\| \geq \|v\|$ for every $g \in G$ (i.e., v has minimum norm in its orbit).
- (ii) If v is critical and $w \in G \cdot v$ is such that $\|v\| = \|w\|$, then $w \in K \cdot v$.
- (iii) If $G \cdot v$ is closed then there exists a critical element $v' \in G \cdot v$.
- (iv) If v is critical then $G \cdot v$ is closed.

In particular, v is a minimum norm vector for itself (i.e., has minimum norm in $\overline{G \cdot v}$) if and only if it has minimum norm in its orbit (meaning $\|g \cdot v\| \geq \|v\|$ for all $g \in G$), which is the case if and only if v is critical.

Thus, minimum norm vectors (or critical vectors) are unique up to the K -action, and their G -orbits are closed, hence provide essentially canonical representatives of orbit closures.

We now comment on the proof of Theorem 2.4.4 as given in [KN79]. Part (iii) follows from non-negativity and continuity of the norm-function with respect to the Euclidean topology on V : the function is bounded from below on $G \cdot v$, assumed to be closed (Zariski-closed implies Euclidean-closed), hence has a minimizer. Parts (i) and (ii) are less trivial. Let us assume that $v \neq 0$. Then consider the Kempf–Ness function $F_v: G \rightarrow \mathbb{R}$ defined by $F_v(g) = \log \|g \cdot v\|$ (Definition 2.4.2). Then F_v is K -invariant, in the sense that $F_v(kg) = F_v(g)$ for every $k \in K$ and $g \in G$. Therefore F_v may also be viewed as a function on the quotient $K \backslash G$. Parts (i) and (ii) then assert that F_v has a unique critical point, which is simultaneously its minimizer. The key reason is that, when $K \backslash G$ is endowed with an appropriate geometry, the function F_v is (strictly) *convex along geodesics*. We shall now use the concrete meaning of this statement for the proof (in fact only for e^{F_v} , which is weaker), and defer a detailed discussion of geodesic convexity of the Kempf–Ness function to Proposition 2.6.6 and Chapters 6 and 10.

We first prove a short proposition.

Proposition 2.4.5. *Let $\Omega \subseteq \mathbb{Z}^n$ be a finite set and let $q_\omega \geq 0$ for $\omega \in \Omega$. Define a function $\alpha: \mathbb{R}^n \rightarrow \mathbb{R}$ by*

$$\alpha(x) = \sum_{\omega} q_\omega e^{2\langle \omega, x \rangle}.$$

Then:

- (i) 0 is a critical point of α if and only if $\alpha(x) \geq \alpha(0)$ for all $x \in \mathbb{R}^n$.
- (ii) If 0 is a critical point of α and $\alpha(x) = \alpha(0)$ for some $x \in \mathbb{R}^n$, then $\langle \omega, x \rangle = 0$ for all $\omega \in \Omega$ such that $q_\omega > 0$.

Proof. (i) This follows directly from the fact that α is a convex function on \mathbb{R}^n .

(ii) Consider the function $b(t) = \alpha(tx)$ for $t \in \mathbb{R}$. By convexity, $b(t) \leq t\alpha(0) + (1-t)\alpha(x) = \alpha(0)$ for $t \in [0, 1]$. Since 0 is critical for α , we also have $b(t) \geq b(x)$ for all $t \in [0, 1]$. Therefore b and its derivative b' are constant on $[0, 1]$. This implies that $b''(0) = 4 \sum_{\omega} q_\omega \langle \omega, x \rangle^2 = 0$, hence either $q_\omega = 0$ or $\langle \omega, x \rangle = 0$. \square

Proof of Theorem 2.4.4. (i). The non-trivial direction is to show that if v is critical, then $\|g \cdot v\| \geq \|v\|$ for every $g \in G$. Fix $g \in G$. By the Cartan decomposition (Theorem 2.2.15), there exist $k, h \in K$ and $t \in T$ such that $g = kth$. Then $g = (kh)(h^{-1}th)$ is such that $kh \in K$ and $h^{-1}th \in h^{-1}Th$, which is a maximal algebraic torus in G and symmetric (since $h \in K$ implies $h^{-1} = h^*$). Choose an adjoint-preserving isomorphism $(\mathbb{C}^\times)^n \cong T$; this yields an adjoint-preserving isomorphism $(\mathbb{C}^\times)^n \cong h^{-1}Th$ as well since $h \in K$. By Theorem 2.2.13, there exists a finite set $\Omega \subseteq \mathbb{Z}^n$ such that $V = \bigoplus_{\omega \in \Omega} V_\omega$ orthogonally decomposes into weight spaces.

Now observe that if $h^{-1}th = (z_1, \dots, z_n)$ under the isomorphism and $v = \sum_{\omega \in \Omega} v_\omega$, we have

$$\|g \cdot v\|^2 = \|((kh)(h^{-1}th)) \cdot v\|^2 = \|(h^{-1}th) \cdot v\|^2 = \sum_{\omega \in \Omega} |z|^{2\omega} \|v_\omega\|^2$$

where $|z|^{2\omega} = (|z_1|^2)^{\omega_1} \cdots (|z_n|^2)^{\omega_n}$. It is now convenient to make a change of coordinates: let $x_j = \log|z_j|$. Then

$$\|g \cdot v\|^2 = \sum_{\omega \in \Omega} e^{2\langle \omega, x \rangle} \|v_\omega\|^2$$

Observe that this function is *convex* in x . Since v is critical, we have $\partial_{s=0} \|e^{sX} \cdot v\|^2 = 0$ for every $X \in \text{Lie}(G)$, so in particular for $X \in \text{Lie}(h^{-1}Th)$. We now invoke Proposition 2.4.5 to obtain that $\|(h^{-1}th) \cdot v\|^2 \geq \|v\|^2$. Item (ii) now also follows: if $\|g \cdot v\| = \|v\|$, then in the weight decomposition for $h^{-1}Th$ we see that for every $\omega \in \Omega$, either $q_\omega = 0$ or $\langle \omega, x \rangle = 0$. This implies that $(h^{-1}th) \cdot v = v$, and so $g \cdot v = kh \cdot (h^{-1}th) \cdot v = kh \cdot v \in K \cdot v$.

We prove part (iv) by contraposition. Suppose that $G \cdot v$ is not closed. Then by the Hilbert–Mumford criterion (Theorem 2.3.16) there exists a point $w \in \overline{G \cdot v} \setminus (G \cdot v)$ and a symmetric one-parameter subgroup $\varphi: \mathbb{C}^\times \rightarrow G$ such that $\lim_{z \rightarrow 0} \varphi(z)v = w$. Again by the weight decomposition, there exist integers $\Omega \subset \mathbb{Z}$ and coefficients q_ω such that

$$\|\varphi(e^x)v\|^2 = \sum_{\omega \in \Omega} q_\omega e^{2\omega x}.$$

Since $\lim_{z \rightarrow 0} \varphi(z)v = w$, we must have

$$\lim_{x \rightarrow -\infty} \|\varphi(e^x)v\|^2 < \infty;$$

hence for every $\omega \in \Omega$, either $q_\omega = 0$ or $\omega \geq 0$. Furthermore, since w is not in $G \cdot v$, there is some $\omega > 0$ with $q_\omega > 0$, and in particular we have $\|\varphi(e^x)v\| < \|v\|$ for $x < 0$. This shows that v is not critical. \square

2.5. The moment map

We saw in the Kempf–Ness theorem (Theorem 2.4.4) that critical vectors have minimal norm among all vectors in their orbit closure, and every orbit closure contains such vectors. Criticality was defined in terms of the derivative of the Kempf–Ness (log-norm) function on the group. This derivative turns out to be interesting in its own right. Let $G \subseteq GL(n, \mathbb{C})$ be a connected symmetric linear algebraic group, $K = G \cap U(n)$ a maximal compact subgroup, and $\pi: G \rightarrow GL(V)$ a regular representation such that V is a Hilbert space and $\pi(K) \subseteq U(V)$.

Definition 2.5.1 (Moment map). For $v \in V \setminus \{0\}$, let $F_v: G \rightarrow \mathbb{R}$, $g \mapsto \log\|g \cdot v\|$ be its Kempf–Ness function. The moment map¹⁹ $\mu: V \setminus \{0\} \rightarrow i\text{Lie}(K)$ is given by

$$\mu(v) = \text{grad}_{g=I} F_v(g) = \text{grad}_{g=I} \log\|g \cdot v\|.$$

The gradient is taken with respect to the Hilbert–Schmidt inner product on $\text{Lie}(G) \subseteq \mathbb{C}^{n \times n}$, i.e., it is uniquely determined by

$$\langle \mu(v), X \rangle = \text{Tr}[\mu(v)^* X] = \partial_{t=0} \log\|e^{tX} \cdot v\| = \partial_{t=0} F_v(e^{tX}), \quad X \in \text{Lie}(G).$$

Some comments on this definition are in order. Although not immediately obvious, the moment map as defined above is actually a moment map in the symplectic sense for the action of K on the projectivization $\mathbb{P}(V)$ of V [NM84], where the projectivization of V is endowed with (a scalar multiple of) the Fubini–Study form. The moment map is also *K-equivariant* with respect to the adjoint action on $i\text{Lie}(K)$, in the sense that $\mu(k \cdot v) = k\mu(v)k^{-1}$ for $k \in K$. Its codomain is also slightly non-standard: usually, $\mu(v)$ would be an element of the dual $\text{Lie}(K)^*$. However, the above concrete definition naturally takes values in $i\text{Lie}(K)$, since K acts unitarily on V and hence the derivative in those directions is zero. As a consequence, $\mu(v) \in i\text{Lie}(K) \subseteq \text{Herm}(n)$ and hence $\mu(v)^* = \mu(v)$.

When $G = T = (\mathbb{C}^\times)^n$ is an algebraic torus, the codomain of the moment map is $i\text{Lie}(T_K) = \mathbb{R}^n$. The image $\mu(T \cdot v)$ is then the set of all possible gradients of the Kempf–Ness function F_v on the quotient space $T_K \backslash T \cong \mathbb{R}^n$, where the isomorphism is given by $(x_1, \dots, x_n) \leftrightarrow (e^{x_1}, \dots, e^{x_n})$. Under this isomorphism, F_v is of the form $\frac{1}{2} \log\left(\sum_{\omega \in \Omega} |v_\omega|^2 e^{2\langle \omega, x \rangle}\right)$, where the $\Omega \subset \mathbb{Z}^n$ is the set of weights appearing in the weight decomposition $V = \bigoplus_{\omega \in \Omega} V_\omega$, and $v = \sum_{\omega \in \Omega} v_\omega$; see Definition 2.2.14.

A straightforward computation yields the following proposition:

Proposition 2.5.2. *Let $\pi: (\mathbb{C}^\times)^n \rightarrow GL(V)$ be an action on V with $\pi(U(1)^n) \subseteq U(V)$ and weights $\Omega \subset \mathbb{Z}^n$. Then for $v \in V \setminus \{0\}$ of the form $v = \sum_{\omega \in \Omega} v_\omega$,*

$$\mu(v) = \frac{\sum_{\omega \in \Omega} |v_\omega|^2 \omega}{\|v\|^2}.$$

Therefore, $\mu(v)$ is in the *convex hull* of the *support* $\text{supp } v \subseteq \Omega$, i.e., the $\omega \in \Omega$ for which $v_\omega \neq 0$. From elementary convex analysis [Roc70, Thm. 26.5] it also follows

¹⁹The name *moment map* as introduced in English by Marsden and Weinstein [MW74] is technically a mistranslation of the French “*application moment*”, a term introduced in [Sou67]. Although the correct choice would be to call it a *momentum map*, we do not break this tradition.

that the image $\mu(T \cdot v)$ of an orbit is an open convex set in $i\text{Lie}(T_K) \cong \mathbb{R}^n$. In fact, its image is entire relative interior of $\text{conv}(\text{supp } v)$, hence its closure (with respect to the Euclidean topology) is $\text{conv}(\text{supp } v)$. We give a self-contained proof of this fact in Chapter 5. More generally, for Hamiltonian torus actions on connected compact symplectic manifolds, the image of the moment map is a convex set [Ati82; GS82; Kir84a].

For non-commutative G , a variation of the convexity statement is also true [Kos73; NM84; GS84; Bri87], even outside the Kähler setting [Kir84a]. A precise statement requires somewhat more terminology. One can choose a closed cone $i\text{Lie}(T_K)_+ \subseteq i\text{Lie}(K)$ called a *positive Weyl chamber*. Then for every $H \in i\text{Lie}(K)$, the adjoint orbit $\{kHk^{-1} : k \in K\}$ intersects $i\text{Lie}(T_K)_+$ in a unique point [Hel79, Ch. VII, Prop. 2.2, Thm. 2.22]. For $K = U(n)$, this amounts to the statement that a Hermitian matrix is unitarily diagonalizable, such that the diagonal elements are ordered in a decreasing manner. We define this to be $\text{spec}_\searrow(H)$.

Definition 2.5.3 (Moment polytope). Let $v \in V \setminus \{0\}$. Then the *moment polytope* $\Delta(v)$ of v is defined as

$$\Delta(v) = \overline{\text{spec}_\searrow(\mu(G \cdot v))} = \overline{\mu(G \cdot v) \cap i\text{Lie}(T_K)_+},$$

where the closure is taken with respect to the Euclidean topology on $i\text{Lie}(T_K)_+ \subseteq i\text{Lie}(T_K)$. The *moment polytope* of V is defined by $\Delta = \bigcup_{v \in V \setminus \{0\}} \Delta(v)$, or equivalently $\Delta = \overline{\text{spec}_\searrow \mu(V \setminus \{0\})}$.

The nomenclature is justified by the following theorem [NM84; GS84; Bri87]:

Theorem 2.5.4. *Let $v \in V \setminus \{0\}$. Then $\Delta(v)$ is a convex polytope.*

The convexity holds in fact not just for orbit (closures) but for arbitrary irreducible closed G -subvarieties of $\mathbb{P}(V)$.

Using the moment map and the various notions of stability (Definition 2.3.17), the Kempf–Ness theorem (Theorem 2.4.4) can be reformulated as follows:

Theorem 2.5.5. *Let $v \in V \setminus \{0\}$. Then:*

- (i) $\mu(v) = 0$ if and only if $\|g \cdot v\| \geq \|v\|$ for every $g \in G$.
- (ii) If $\mu(v) = 0$, then $\mu^{-1}(0) \cap G \cdot v = K \cdot v$.
- (iii) If v is polystable then $0 \in \mu(G \cdot v)$.
- (iv) If $\mu(v) = 0$ then v is polystable.

Furthermore, v is semistable if and only if $0 \in \Delta(v)$.

We note here that the above rephrasing has the following interesting consequence. Consider the map $\tilde{\mu}: \mathbb{P}(V) \rightarrow i\text{Lie}(K)$ given by $\tilde{\mu}([v]) = \mu(v)$ (note that μ is invariant under rescaling v). Assume also that $0 \in i\text{Lie}(K)$ is a *regular value* of $\tilde{\mu}$, and that K acts freely and properly on $M = \tilde{\mu}^{-1}(0)$. Then the *symplectic quotient* [MW74; Mey73] $\tilde{\mu}^{-1}(0)/K$ is a symplectic manifold whose points are in one-to-one correspondence with the GIT quotient $\mathbb{P}(V)^{ss}/G$, and this fact is of great importance. We refer to [Kir98] for a modern survey.

Next, we give an example of a non-commutative moment map:

2. Setting the stage

Example 2.5.6 (Adjoint action). Let $G = \text{GL}(n, \mathbb{C})$ act on $V = \mathbb{C}^{n \times n}$ via conjugation. We endow V with the Hilbert–Schmidt norm. Then for $v \in V \setminus \{0\}$ the Kempf–Ness function is given by $F_v(g) = \log \|gvg^{-1}\|_{\text{HS}}$. Given a Hermitian matrix $H \in \mathfrak{iLie}(U(n))$, we have

$$\begin{aligned} \partial_{t=0} F_v(e^{tH}) &= \log \|e^{tH} v e^{-tH}\|_{\text{HS}} = \frac{1}{2} \frac{\partial_{t=0} \text{Tr}[e^{-tH} v^* e^{2tH} v e^{-tH}]}{\|v\|_{\text{HS}}^2} \\ &= \frac{\text{Tr}[H(vv^* - v^*v)]}{\|v\|_{\text{HS}}^2}. \end{aligned}$$

Since the moment map $\mu(v)$ is characterized by $\text{Tr}[\mu(v)H] = \partial_{t=0} F_v(e^{tH})$, we obtain

$$\mu(v) = \frac{vv^* - v^*v}{\|v\|_{\text{HS}}^2}.$$

Therefore v is critical if and only if v is unitarily diagonalizable, and v is polystable if and only if it becomes unitarily diagonalizable after some (non-unitary) change of basis, thus if and only if v is similar to a diagonal matrix, as we saw in Example 2.3.18.

Example 2.5.7 (Bipartite quantum states). Let $G = \text{GL}(n, \mathbb{C}) \times \text{GL}(n, \mathbb{C})$ act on $V = \mathbb{C}^n \otimes \mathbb{C}^n$ by $g \cdot v = (g_1 \otimes g_2)v$. Then for $H = (H_1, H_2) \in \mathfrak{iLie}(K) = \text{Herm}(n) \oplus \text{Herm}(n)$ a tuple of Hermitian matrices, we have

$$\begin{aligned} \partial_{t=0} \log \|\exp(tH) \cdot v\| &= \partial_{t=0} \log \|(e^{tH_1} \otimes e^{tH_2})v\| \\ &= \frac{1}{\|v\|^2} (\langle v, (H_1 \otimes I)v \rangle + \langle v, (I \otimes H_2)v \rangle) \\ &= \text{Tr}[H_1 \rho_1] + \text{Tr}[H_2 \rho_2] = \langle (H_1, H_2), (\rho_1, \rho_2) \rangle, \end{aligned}$$

where $\rho = \frac{|v\rangle\langle v|}{\langle v|v\rangle}$ and $\rho_1 = \text{Tr}_2[\rho]$, $\rho_2 = \text{Tr}_1[\rho]$ are its partial traces. Therefore the moment map is given by

$$\mu(v) = (\rho_1, \rho_2).$$

Now the eigenvalues of the first component are exactly given by the Schmidt coefficients of v (its singular values when viewed as an operator $\mathbb{C}^n \rightarrow \mathbb{C}^n$), and the same holds for the second component. Clearly, the number of non-zero coefficients cannot increase by acting with G , but the Schmidt coefficients can be changed arbitrarily (by acting with diagonal matrices in the Schmidt basis). Therefore $\mu(G \cdot v)$ consists of $(\sigma_1, \sigma_2) \in \text{Herm}(n) \times \text{Herm}(n)$ with σ_1, σ_2 positive semidefinite, having the same eigenvalues, $\text{Tr}[\sigma_1] = \text{Tr}[\sigma_2] = 1$, and $\text{rank}(\sigma_1) \leq \text{rank}(\rho_1)$. Note that all these constraints can be viewed as *linear* inequalities on the ordered spectra of ρ_1, ρ_2 , hence the achievable ordered spectra form a convex polytope.

The moment polytope also admits a purely representation-theoretic description. If $\mathbb{C}[V]_d$ denotes the ring of homogeneous polynomials of degree d on V , then $\mathbb{C}[V]_d$ is also a representation of G , under $(g \cdot p)(v) = p(g^{-1} \cdot v)$. The irreducible representations V_λ of G are determined by their *highest weights* $\lambda \in \text{Lie}(T_K)_+$, which is a *dominant integral element* of $\text{Lie}(T)^*$. In the case of $\text{GL}(n, \mathbb{C})$, the irreducible representations are labelled by integer sequences $\lambda_1 \geq \dots \geq \lambda_n$. If V_λ denotes the irreducible representation labelled by a dominant integral vector λ , then the following was shown by Mumford [NM84] and Brion [Bri87]:

Theorem 2.5.8. *The moment polytope Δ satisfies*

$$\Delta = \overline{\left\{ \frac{\lambda}{d} : V_\lambda \text{ is a subrepresentation of } \mathbb{C}[V]_d \right\}}.$$

A similar description exists for the moment polytope of an irreducible closed G -subvariety of $\mathbb{P}(V)$, where $\mathbb{C}[V]_d$ is replaced by the homogeneous functions of degree d on the cone over X . This representation-theoretic characterization yields the connection between the quantum marginal problem and the asymptotic non-vanishing of Kronecker coefficients, see e.g. [Kly04].

2.6. The computational problems

We now put a more computational spin on the theory developed in the previous sections. Let $G \subseteq \mathrm{GL}(n, \mathbb{C})$ be a connected symmetric linear algebraic group, and $\pi: G \rightarrow \mathrm{GL}(V)$ a regular representation such that V is endowed with a K -invariant inner product $\langle \cdot, \cdot \rangle$, and induced norm $\|\cdot\|$. Suppose we are given explicit descriptions of G as a subgroup of $\mathrm{GL}(n, \mathbb{C})$ described by finitely many polynomial equations, the representation π described by its matrix coefficients with respect to some basis of V , and the vectors $v, v' \in V$ in the same basis. How does one decide *algorithmically* whether $\overline{G \cdot v} \cap \overline{G \cdot v'} \neq \emptyset$? We shall refer to this as the *orbit closure intersection* (OCI) problem.

Problem 2.6.1 (Orbit closure intersection). *Given a linearly reductive group $G \subseteq \mathrm{GL}(n, \mathbb{C})$, a regular representation $\pi: G \rightarrow \mathrm{GL}(V)$, and vectors $v, v' \in V$, determine whether $\overline{G \cdot v} \cap \overline{G \cdot v'} \neq \emptyset$.*

Mumford's theorem (Theorem 2.3.7) together with Hilbert's finiteness theorem (Theorem 2.3.12) suggest that this is a decidable problem. In fact, there exist algorithms that, given equations for G and the entries of the representation π expressed in some basis of V , compute generators $p_1, \dots, p_r \in \mathbb{C}[V]^G$ [DK15]. Accordingly, determining whether two vectors v, v' are equivalent in the sense of GIT (i.e., $\overline{G \cdot v} \cap \overline{G \cdot v'} \neq \emptyset$) can in principle be decided by an algorithm – simply check whether $p_j(v) = p_j(v')$ for all $j \in [r]$. However, this is impractical, since known algorithms for computing generators are inefficient (run in exponential time or worse) and in many situations one will have to deal with generators that have exponentially large degree (we will in fact see an explicit example in Section 3.3) or are hard to evaluate in the sense of computational complexity [GIM+20]. Furthermore, it is not clear how such an algebraic approach could go beyond the decision problem to compute, e.g., an element in the orbit closure intersection, or a sequence of group elements that drives one there. We note here that in the commutative setting one also has these obstructions, but here one can still use an invariant-theoretic approach to solve the OCI problem [BDM+21]; for OCI in the setting of operator scaling one can also give polynomial-time algorithms based on invariants [DM20a].

We have now seen in the Kempf–Ness theorem (Theorem 2.4.4) that minimum norm vectors provide essentially canonical representatives of orbit closures.

2. Setting the stage

Therefore, an alternative approach as follows: to decide whether v and v' have intersecting orbit closures, first compute minimum norm vectors v_{\min} and v'_{\min} , and then determine whether there exists a $k \in K$ such that $v_{\min} = k \cdot v'_{\min}$. Of course, this is easier said than done, although it has been done in the setting of operator scaling [AGL+18]. One should in general not be expect to be able to compute the minimum norm vectors v_{\min} *exactly*; for example, they may have irrational entries,²⁰ so rational-number arithmetic can only produce approximate results. It is also not obvious how to certify that a vector has *approximately* minimal norm in its orbit closure. Moreover, it may be computationally difficult to decide whether two vectors are in the same K -orbit; this problem is known to be at least as hard as graph isomorphism [CGQ+23], but efficiently solvable in special cases [AGL+18; BDM+21; DKMV23].

In light of the notion of (in)stability (Section 2.3.4), one particularly interesting case of the OCI problem is when $v' = 0$, i.e., determining whether v is in the null cone.

Problem 2.6.2 (Null cone). *Given a regular representation $\pi: G \rightarrow GL(V)$ of a connected linearly reductive group $G \subseteq GL(n, \mathbb{C})$, and $v \in V \setminus \{0\}$, determine whether $0 \in \overline{G \cdot v}$.*

We now observe that by the Kempf–Ness theorem, $v \in V$ is in the null cone if and only if $v_{\min} = 0$, which holds if and only if $\inf_{g \in G} \|g \cdot v\| = 0$. Therefore, the null cone problem can be viewed as characterizing the optimal value of the norm minimization problem:

Problem 2.6.3 (Norm minimization). *Given a regular representation $\pi: G \rightarrow GL(V)$ of a connected linearly reductive group $G \subseteq GL(n, \mathbb{C})$, $v \in V \setminus \{0\}$ and $\delta > 0$, output either $g \in G$ such that $\log \|g \cdot v\| \leq \log \|v_{\min}\| + \delta$, or assert that v is in the null cone of V .*

Rephrased in terms of the Kempf–Ness function $F_v(g) = \log \|g \cdot v\|$, the norm minimization problem asks to find a δ -approximate minimizer of F_v , or to assert that F_v is unbounded from below. Note that taking the logarithm of the norm is natural given the scale invariance of the problem: if v_{\min} is a minimum norm vector for v , then λv_{\min} is a minimum norm vector for λv .

There is another natural error measure: recall from Theorem 2.4.4 that v is a minimum norm vector if and only if v is critical, i.e., $\mu(v) = 0$ where $\mu(v) = \text{grad}_{g=I} F_v(g)$ is the moment map and F_v is the Kempf–Ness function. Therefore the *norm* of the moment map is also a natural error measure, as this measures the distance from zero; we use here the norm induced by the Hilbert–Schmidt norm on $\text{Lie}(G) \subseteq \mathbb{C}^{n \times n}$.

Problem 2.6.4 (Scaling). *Given a regular representation $\pi: G \rightarrow GL(V)$ of a connected linearly reductive algebraic group $G \subseteq GL(n, \mathbb{C})$, $v \in V \setminus \{0\}$ and $\varepsilon > 0$, output either $g \in G$ such that $\|\mu(g \cdot v)\| \leq \varepsilon$, or assert that v is in the null cone of V .*

The norm-minimization and scaling problems are equivalent for $\delta = \varepsilon = 0$ as shown by the Kempf–Ness theorem. Although not obvious, it turns out a *quantitative* relationship also holds [BFG+19]. To state this, we define the following two parameters:

²⁰A simple example is given by \mathbb{C}^\times acting on \mathbb{C}^2 via $z \cdot (v_1, v_2) = (zv_1, z^{-1}v_2)$. The $U(1)$ -orbit of minimum norm vectors of $v = (1, 2)$ is given by $U(1) \cdot (\sqrt{2}, \sqrt{2})$.

Definition 2.6.5 (Weight margin and weight norm). The *weight margin* $\gamma(\pi)$ of the representation π is defined as

$$\gamma(\pi) = \min\{d(0, \text{conv } \Gamma) : \Gamma \subseteq \Omega(\pi), 0 \notin \text{conv } \Gamma\}.$$

Here, $\text{conv } \Gamma$ refers to the convex hull of $\Gamma \subseteq \text{Lie}(T)^*$. The *weight norm* $N(\pi)$ is defined by

$$N(\pi) = \max\{\|\omega\| : \omega \in \Omega(\pi)\}.$$

The distance $d(\cdot, \cdot)$ and $\|\cdot\|$ are defined in terms of the Hilbert-Schmidt inner product after identifying $\text{Lie}(T)^* \cong \text{Lie}(T) \subseteq \mathbb{C}^{n \times n}$.

While these parameters are somewhat abstract, we give a short justification for their appearance. Let $v \in V \setminus \{0\}$, and restrict to the case where $G = T = (\mathbb{C}^\times)^n$ is commutative. Recall from Proposition 2.5.2 that if one considers the weights $\Omega(\pi)$ as a subset of $i\text{Lie}(T_{\mathbb{K}})$ (Definition 2.2.14) and $v = \sum_{\omega \in \Omega(\pi)} v_\omega$, then one has

$$\mu(v) = \frac{1}{\|v\|^2} \sum_{\omega \in \Omega} \|v_\omega\|^2 \omega.$$

We observe now that the support $\text{supp } v$, i.e., the set of ω such that $v_\omega \neq 0$, does not change when one acts with $G = T$. This implies that if $v \in V \setminus \{0\}$ is such that the convex hull of its support does not contain 0, then $\|\mu(g \cdot v)\| \geq \gamma(\pi)$ for all $g \in G$. Note that this also implies that $v_{\min} = 0$: one can use a (rational) separating hyperplane between 0 and $\text{conv}(\text{supp } v)$ to find a direction $Y \in i\text{Lie}(K)$ such that $e^{tY} \cdot v \rightarrow 0$ as $t \rightarrow \infty$. This is exactly the Hilbert–Mumford criterion (Theorem 2.3.16) in the unstable case for commutative groups.

In the non-commutative case, for $T \subseteq G$ a maximal algebraic torus, the moment map μ_T with respect to T is the *projection* of μ_G onto $i\text{Lie}(T_{\mathbb{K}})$, and so $\|\mu_G(v)\| \geq \|\mu_T(v)\|$. By the Hilbert–Mumford criterion, if a vector v is G -unstable, then it is T -unstable with respect to *some* T . Hence $\|\mu_G(v)\| \geq \|\mu_T(v)\| \geq \gamma(\pi)$. Therefore also in the non-commutative case it is true that $\|\mu_G(v)\| < \gamma(\pi)$ implies that v is semistable.

The appearance of the weight norm $N(\pi)$ is simpler to explain: it is an upper bound on the norm of the moment map $\mu(v)$. But $\mu(g \cdot v)$ is the gradient of the Kempf–Ness function F_v at $g \in G$, and hence F_v is $N(\pi)$ -Lipschitz (thought of as a function on $K \backslash G$). More generally $N(\pi)$ appears when bounding the higher-order derivatives of F_v [BFG+19, Prop. 3.13]:

Proposition 2.6.6 (Smoothness). *Let $v \in V \setminus \{0\}$. Then for every $g \in G$ and $H \in i\text{Lie}(K)$,*

$$0 \leq \partial_{t=0}^2 F_v(e^{tH}g) \leq 2N(\pi)^2 \|H\|_{\text{HS}}^2.$$

Proof. Without loss of generality one may take $g = I$, since $F_v(e^{tH}g) = F_{g \cdot v}(e^{tH})$. Recall that we write $\langle \cdot, \cdot \rangle$ for the inner product on V . Let $\Pi = d\pi_I : \text{Lie}(G) \rightarrow \text{Lie}(\text{GL}(V))$ be the induced representation on the Lie algebras. Then

$$\begin{aligned} \partial_t F_v(e^{tH}) &= \frac{1}{2} \partial_t \log \langle \pi(e^{tH})v, \pi(e^{tH})v \rangle \\ &= \frac{\langle \pi(e^{tH})v, \Pi(H)\pi(e^{tH})v \rangle}{\|\pi(e^{tH})v\|^2} \end{aligned}$$

2. Setting the stage

and

$$\begin{aligned} \partial_{t=0}^2 F_v(e^{tH}) &= 2 \frac{\langle \Pi(H)v, \Pi(H)v \rangle}{\|v\|^2} - 2 \frac{\langle v, \Pi(H)v \rangle^2}{\|v\|^4} \\ &= 2 \left(\langle \Pi(H)w, \Pi(H)w \rangle - \langle w, \Pi(H)w \rangle^2 \right) \end{aligned} \quad (2.6.1)$$

$$= 2 \langle w, (\Pi(H) - \langle w, \Pi(H)w \rangle I)^2 w \rangle. \quad (2.6.2)$$

where $w = v/\|v\|$. Now Π maps $i\text{Lie}(K)$ to $i\text{Lie}(U(V)) \subseteq \text{Herm}(V)$ and hence $\Pi(H) - \langle w, \Pi(H)w \rangle I$ is a Hermitian operator, hence its square is a positive-semidefinite operator. In particular, Eq. (2.6.2) gives $\partial_{t=0}^2 F_v(e^{tH}) \geq 0$. Moreover, using Eq. (2.6.1) we can upper bound the second derivative by

$$2\|\Pi(H)w\|^2 \leq 2\|\Pi(H)\|_\infty^2 \leq 2N(\pi)^2\|H\|_{\text{HS}}^2$$

where $\|\cdot\|_\infty$ is the operator norm. For the last inequality, we appeal to [BFG+19, Prop. 3.11], which states that the weight norm $N(\pi)$ is equal to $\max\{\|\Pi(H)\|_\infty : H \in i\text{Lie}(K), \|H\|_{\text{HS}} = 1\}$. This fact can be proven by explicitly considering a maximal torus whose Lie algebra contains H . \square

We are now in a position to state a quantitative relationship between the norm of the moment map and the approximation ratio $\|v_{\min}\|/\|v\|$ for $v \in V \setminus \{0\}$:

Theorem 2.6.7 (Non-commutative duality, [BFG+19, Thm. 1.17]). *For $v \in V \setminus \{0\}$ with minimum norm vector v_{\min} (Definition 2.4.1), we have*

$$1 - \frac{\|\mu(v)\|}{\gamma(\pi)} \leq \frac{\|v_{\min}\|^2}{\|v\|^2} \leq 1 - \frac{\|\mu(v)\|^2}{4N(\pi)^2}.$$

The upper bound can be deduced from a geodesic gradient descent argument for F_v (Proposition 6.5.3), using the $2N(\pi)^2$ -smoothness proved in Proposition 2.6.6. The lower bound is more delicate to prove, and we do not comment further on it.

One can generalize the scaling problem as follows: rather than just ask whether 0 is in the moment polytope, one can also ask if a specific (rational) point $p \in i\text{Lie}(T_K)_+$ is in the moment polytope $\Delta(v)$ of a given vector $v \in V \setminus \{0\}$. We do not state this problem formally here and refer instead to [BFG+19]. However, we note that this is still related to a (geodesic) convex optimization problem. In the commutative case, one can simply modify the Kempf–Ness by simply adding a linear function (after a change of coordinates), see Chapter 5 for details. In the non-commutative case, the corresponding norm minimization problem is on a different (possibly much larger) representation, as can be seen by a *shifting trick* [NM84; Bri87].

As an extension of the scaling problem, one can also ask the following question: suppose that $v \in V \setminus \{0\}$ is unstable. Then still $\Delta(v)$ is a convex polytope, so there exists a *closest point* to 0, i.e., the projection of 0 onto $\Delta(v)$. Can one find this point efficiently? As was observed in [NM84; Kir84b], following the gradient flow of the function $\mathbb{P}(V) \rightarrow \mathbb{R}$ given by $[v] \mapsto \|\mu(v)\|_{\text{HS}}^2$ from a starting point $[v_0]$ is in a sense equivalent to following the gradient flow of the Kempf–Ness function F_{v_0} . This gradient flow always stays in the G -orbit of $[v_0] \in \mathbb{P}(V)$, and actually converges to a minimizer on its projective orbit closure. This result has been proposed as an algorithmic tool for testing moment polytope membership in [WDGC13; Wal14],

but giving rigorous guarantees for algorithmically following this gradient flow in the unstable case remains an open problem. However, we do note that it has proven useful for the purpose of giving diameter bounds on approximate minimizers for the norm minimization and scaling problems, see for instance [KLLR18] (where it was used to solve the Paulsen problem), [AGL+18; KLR19] and [BFG+19, Prop. 5.6].

3. The minimal canonical form of a tensor network

In this chapter we identify a novel application of geometric invariant theory as discussed in Chapter 2 in the context of tensor networks in quantum many-body physics. The chapter is organized as follows. We provide a detailed introduction below in Section 3.1. In Section 3.2, we show how to apply geometric invariant theory to construct the minimal canonical form for matrix product states. Our main results are proved in Section 3.3, where we introduce the minimal canonical form for PEPS and establish its properties. In Section 3.4 we provide explicit algorithms for computing the minimal canonical form. We end with a brief outlook in Section 3.5, suggesting applications for the minimal canonical form and avenues for future research.

3.1. Introduction

Tensor networks are a fruitful area of interconnection between quantum information theory and quantum many-body physics. On the one hand, tensor network states are rich enough to approximate with high accuracy most states which are relevant in condensed matter physics, such as Gibbs states and ground states. On the other hand, tensor networks are sufficiently simple that they enable one to manipulate complex quantum states, both numerically and theoretically. For the purpose of numerics, one can design variational optimization algorithms to simulate strongly interacting quantum systems. On the other side of the spectrum, tensor networks have been a powerful theoretical method to obtain simple characterizations of complex global phenomena like topological order.

Roughly speaking a tensor network is defined by a set of tensors with two types of indices: virtual ones, whose dimension is called the *bond dimension*, and physical ones, associated to the different subsystems of a quantum many-body system. These tensors generate a state (called a tensor network state) in the physical Hilbert spaces of the system by contracting the virtual indices on a given graph, typically a lattice associated to the interaction pattern of a Hamiltonian. The graphical notation for tensor network contractions is briefly reviewed in Fig. 3.1a.

The success of tensor network states as a numerical variational family dates back to the pioneering paper [Whi92], where the Density Matrix Renormalization Group (DMRG) algorithm was proposed as a way to approximate ground states of one-dimensional systems. Nowadays, this algorithm is seen as a way to minimize energy over the manifold of *Matrix Product States* (MPS), the first and most well-known family of tensor networks. From the perspective of quantum information theory, one may also see MPS as pairs of maximally entangled states to which

This chapter is adapted from [AMN+23].

3. The minimal canonical form of a tensor network

locally a projection operator is applied. This allowed the generalization of the construction to more complex scenarios, including higher dimensions [VPC04; VC04]. There, the associated objects are called *Projected Entangled Pair States* (PEPS), precisely due to the perspective of applying projectors to a configuration of maximally entangled states. By now, there can be no doubt that this is one of the most important and powerful paradigms in numerical simulation of quantum systems [JCF+21; RBC21; SDC+22; ZCC+17], a recent highlight being the classical simulation [PZ22] of the Google *quantum supremacy experiment* [AAB+19].

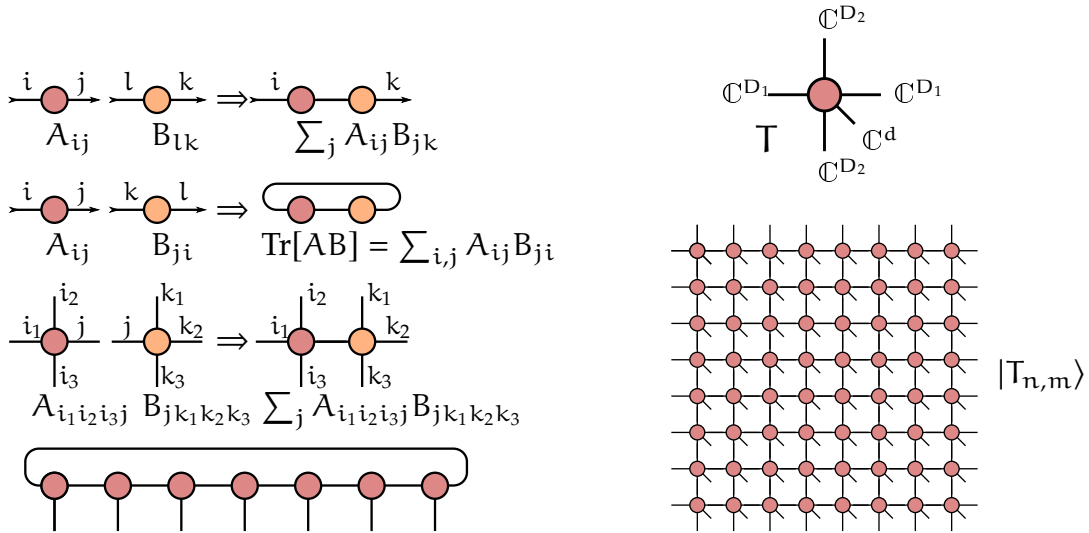
On the theoretical side tensor networks allow one to give local characterizations, in terms of their defining tensors, of global properties of interest, such as symmetries or topological order. The pioneering work [FNW92], independently from the DMRG proposal [Whi92], started this line of research. One of the first milestones was the cohomology-based classification of one-dimensional symmetry-protected topological (SPT) phases [CGW11; PBTO12; SPC11]. Today, this is an active area of investigation, see for instance the recent review [CPSV21] for details on the current state of the art. For instance, tensor networks are used for the characterization of topological order and topological phase transitions in higher spatial dimensions. Other important theoretical results concern rigorous approximation bounds, showing rigorously that classes of physically relevant states such as ground states and Gibbs states can be approximated accurately by PEPS.

Recently, due to their nice numerical and analytical properties, tensor networks have started to permeate other areas. Prominent examples are quantum gravity [HNQ+16; PYHP15] and machine learning [SS16; CPZ+17], as well as (hybrid) classical simulation of quantum circuits [PHOW20; NLD+22].

An important feature both in theory and practice is the **gauge symmetry** of a tensor network. By inserting matrices on the virtual bonds of a tensor in such way that they cancel when the network is contracted, one modifies the local tensors while leaving the many-body state unchanged, see Fig. 3.2a. In this context one desires: (1) a **fundamental theorem** that guarantees the gauge symmetry is the *only* freedom in tensors to give rise to the same states, and (2) a **canonical form**, which *fixes* this gauge degree of freedom in a natural way. Sometimes, both come together: some fundamental theorems only apply to tensors in a canonical form.

To make this more concrete, we consider PEPS in one spatial dimension, i.e., MPS. One key reason which make MPS easier to work with than, e.g., 2D PEPS, is that there are canonical forms with good theoretical properties and an associated fundamental theorem. This has played a crucial role in the development of the theory since its inception [FNW92], see [CPSV21] for a review. We focus on the *uniform* (or *translation-invariant*) case, where one places the same 3-tensor T on each site and contracts with periodic boundary conditions, resulting in a many-body quantum state $|T_n\rangle$ for any system size n . One may view T as a tripartite quantum state on one physical and two virtual Hilbert spaces, the latter of bond dimension D . It is always possible (after blocking sites together and setting irrelevant off-diagonals to zero) to choose a gauge such that the reduced state on one of the two virtual Hilbert space is maximally mixed.¹ The result is called a *left or right canonical form* and it is unique up to *unitary* gauge symmetries. It has the

¹As we will see in Definition 3.2.5, strictly speaking this is only true independently in each of the diagonal blocks which remain in the canonical form. There is a proportionality constant that can be different in each one of those blocks.



(a) *Graphical notation:* A reminder of the graphical notation for tensor network contractions. If the tensors are interpreted as matrices, arrows indicate the direction of multiplication. The examples include matrix multiplication, the trace of a product of matrices, and in the bottom row, a matrix product state.

(b) *Uniform PEPS in 2D:* A tensor T gives rise to states $|T_{n,m}\rangle$ on a periodic $n \times m$ lattice by placing T at the sites and contracting with periodic boundary conditions.

Figure 3.1.

following virtues:

- (A) It satisfies a **fundamental theorem**: two tensors T and T' give rise to the same states on any number of sites, meaning $|T_n\rangle = |T'_n\rangle$ for all n , if and only if they have a common canonical form.
- (B) It allows **lifting symmetries**: if T is in canonical form, any global symmetry $U^{\otimes n} |T_n\rangle = |T_n\rangle$ for all n can be implemented by a *unitary* gauge symmetry on T . This is key to classifying phases of matter and when studying entanglement spectra/Hamiltonians, to upgrade virtual to physical degrees of freedom.
- (C) It provides a way to **truncate**, which is key for efficient accurate numerics: given a tensor T with bond dimension D , it allows finding a tensor T' of bond dimension $D' < D$ such that $|T'_n\rangle \approx |T_n\rangle$ for all n .

Clearly, it would be of great use to extend the theory of canonical forms to tensor networks in two or more spatial dimensions! However, it is known that there are significant obstructions. For example [SMG+20; Sch20]:

- (ζ) The following problem is **undecidable**: Given a PEPS tensor T , decide if the associated states $|T_{n,m}\rangle$ vanish on periodic lattices of any size $n \times m$.

This suggests there should not exist any useful (computable) canonical form generalizing (A), since by comparing the canonical forms of T and the zero tensor

3. The minimal canonical form of a tensor network

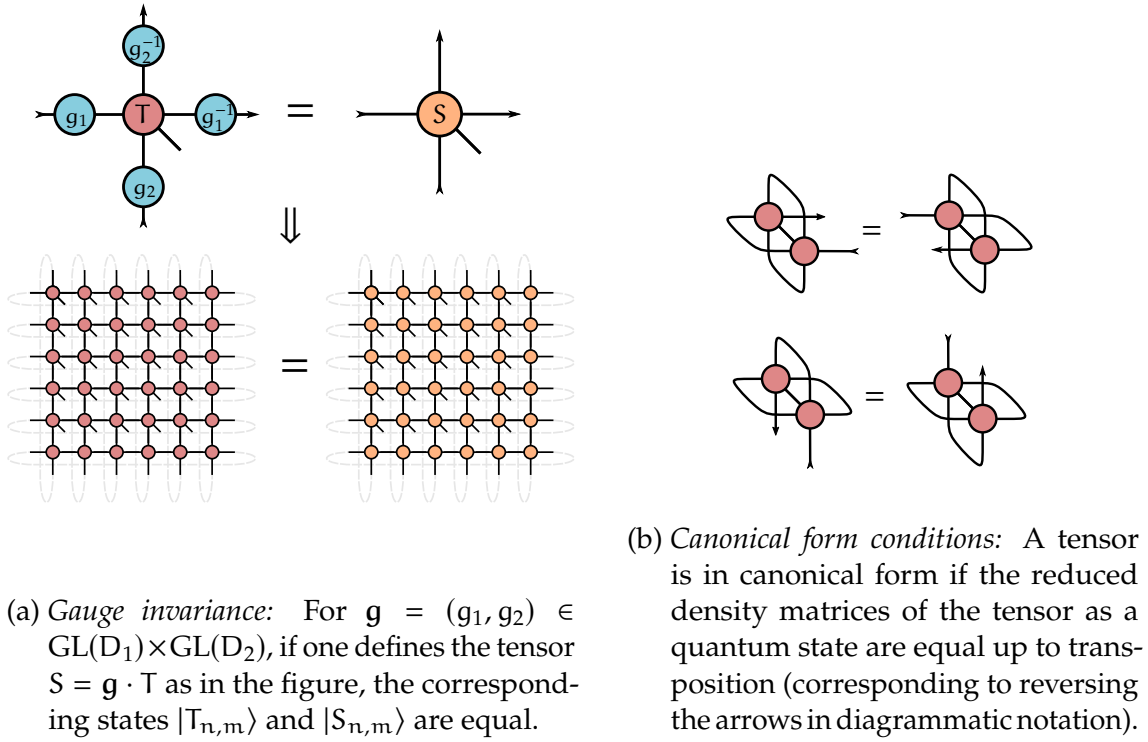


Figure 3.2.

one could otherwise decide whether $|T_{n,m}\rangle = 0$ for all n and m . Indeed, before our work, no canonical form was known for PEPS tensor networks in two or more dimensions that applied to general tensors and rigorously satisfied properties such as the above.

On the other hand, a fundamental theorem is known if one restricts, e.g., to the class of *normal* tensors [MGP+18]. Moreover, heuristic approaches for canonical forms [Eve18; KKOS12; LCB14b; PMV15; PBT+15] and the truncation problem (C) are successfully used in practice to trade off efficient computation and approximation accuracy [RTP+20].

3.1.1. Summary of results: a canonical form in any dimension and a fundamental theorem

In this work we introduce a new canonical form for general PEPS in arbitrary spatial dimension. It rigorously satisfies a number of desirable properties – particularly a *fundamental theorem*. The obstruction (ζ) is overcome by the following twist: roughly speaking, the canonical form captures when two tensors give rise to the same quantum states not just on the torus, but on any surface! This is achieved by pioneering the application of geometric invariant theory, an area of mathematics that studies symmetries, to tensor network theory and drawing on recent research in *non-commutative group optimization*.²

We now define the new canonical form and highlight its main properties

²Geometric invariant theory has already been used in quantum information in other contexts, such as in the study of multipartite entanglement [Kly02; VDD03; GW10; BRV18], or in the quantum marginal problem [Kly04; DH05; CM06; Kly06; WDC13; Wal14], but not in the area of tensor networks to the best of our knowledge.

and the new fundamental theorem. Here we only discuss uniform PEPS in m spatial dimensions. These are defined by a single tensor T , with $2m + 1$ legs, one associated to the physical Hilbert space, and two legs each for the spatial directions $k \in \{1, \dots, m\}$, associated with virtual Hilbert spaces of bond dimension D_k . The gauge group $G = GL(D_1) \times \dots \times GL(D_m)$ acts on the virtual legs of the tensor as illustrated in Fig. 3.2a. We say T_{\min} is a **minimal canonical form** of T if it “infimizes” the ℓ^2 -norm among all gauge equivalent tensors:

$$T_{\min} = \operatorname{argmin} \left\{ \|S\| : S \in \overline{G \cdot T} \right\}. \quad (3.1.1)$$

In the language of Section 2.4, T_{\min} is a minimum norm vector. Two important remarks are in order: First, we consider the closure $\overline{G \cdot T}$ of the gauge group orbit of T , so that the minimum is attained. Thus there need not be a single gauge transformation $\mathbf{g} \in G$ such that $\mathbf{g} \cdot T = T_{\min}$, but rather a sequence $\mathbf{g}^{(k)} \in G$ such that $\mathbf{g}^{(k)} \cdot T \rightarrow T_{\min}$ (the same is true for the usual canonical forms of MPS when one has to set off-diagonal blocks to zero). This is, however, natural, since the uniform PEPS determined by a tensor depend continuously on the tensor, hence remain unchanged even when taking limits. Second, while any tensor clearly has a minimal canonical form, uniqueness up to unitaries is a priori unclear. This is addressed by our first result, which justifies calling T_{\min} a ‘canonical form’.

Result 1 (Canonical form). *Any tensor has a minimal canonical form. It is unique up to unitary gauge symmetry. Moreover, two tensors T, T' have a common minimal canonical form if and only if $\overline{G \cdot T} \cap \overline{G \cdot T'} \neq \emptyset$.*

The condition $\overline{G \cdot T} \cap \overline{G \cdot T'} \neq \emptyset$ is the natural definition of *gauge equivalence*, since then T, T' determine the same PEPS as explained above. Result 1, which we formally state as Theorem 3.2.9 for MPS and Theorem 3.3.7 for PEPS, states that this is captured by the minimal canonical form. It also guarantees the analogue of property (B) for normal tensors, stated as Corollary 3.3.9.

We can characterize the minimal canonical form in terms of the reduced states of the virtual bonds. To this end, interpret T as a quantum state and denote by $\rho_{k,1}$ and $\rho_{k,2}$ the reduced states of the two virtual bonds in the k -th direction. Then we have the following characterization, illustrated in Fig. 3.2b.

Result 2 (Characterization). *A tensor is in minimal canonical form if and only if $\rho_{k,1} = \rho_{k,2}^T$ for $1 \leq k \leq m$.*

Interestingly, this shows our minimal canonical form does *not* coincide with the usual ones for MPS ($m = 1$); it also differs from previously proposed heuristics in higher dimensions. We prove Result 2 in Theorem 3.2.10 for MPS and Theorem 3.3.8 for PEPS.

This begs the question whether it can be computed effectively, even for MPS. Our next result answers this in the affirmative.

Result 3 (Computation). *There is an algorithm which computes a minimal canonical form of a tensor T up to given ℓ^2 -error $\delta > 0$. For fixed bond dimensions, it runs in time polynomial in $\log \frac{1}{\delta}$ and in the bitsize of T .*

3. The minimal canonical form of a tensor network

We prove this in Corollary 3.4.12. The algorithm depends exponentially on the bond dimensions (for $m > 1$). We also give an algorithm whose runtime depends only polynomially on the bond dimension, but also on $\frac{1}{\varepsilon}$, where ε measures the accuracy to which the condition in Result 2 is fulfilled (see Corollary 3.4.3). In Section 3.4 we discuss these and another natural way of quantifying approximation error; we relate them in Section 3.4.2.

Finally, we discuss our fundamental theorem. We start with the following observation (for simplicity in 2D): If two tensors are gauge equivalent, they not only determine the same state $|\mathbb{T}_{n,m}\rangle$ on any $n \times m$ lattice, but also if we contract according to an arbitrary graph such that only left and right virtual legs, and only top and bottom virtual legs are connected. We say Γ is a *contraction graph* and write $|\mathbb{T}_\Gamma\rangle$ for the corresponding uniform PEPS, see Fig. 3.3. Intuitively, this means we consider tensor networks on surfaces of *arbitrary topology* rather than only on the torus. Clearly, these notions generalize to any spatial dimension. We find that this precisely captures gauge equivalence, in any spatial dimension! Indeed, we have the following result which we formalize and prove as Theorem 3.3.11:

Result 4 (Fundamental theorem). *Two tensors \mathbb{T}, \mathbb{T}' are gauge equivalent (meaning $\overline{\mathbb{G} \cdot \mathbb{T}} \cap \overline{\mathbb{G} \cdot \mathbb{T}'} \neq \emptyset$) if and only if $|\mathbb{T}_\Gamma\rangle = |\mathbb{T}'_\Gamma\rangle$ for all contraction graphs Γ . It suffices to consider to graphs on $e^{\tilde{O}(mD^2)}$ vertices.*

We further show $e^{\Omega(mD)}$ vertices are necessary when $m \geq 2$, while for $m = 1$ we find two MPS tensors to be gauge equivalent iff $|\mathbb{T}_n\rangle = |\mathbb{T}'_n\rangle$ for $1 \leq n \leq \tilde{O}(D)$, which is essentially tight [DM20a]. While we stress that our fundamental theorem is of independent interest, as it precisely characterizes when two tensors are gauge equivalent, we note that gauge equivalence is the same as having a common canonical form (by Result 1). Accordingly, our theorem proves a version of property (A) for PEPS in any spatial dimension, and as we show in Corollary 3.3.14, this also implies global symmetries of the states $|\mathbb{T}_\Gamma\rangle$ can be lifted to unitary gauge symmetries, as in property (B). Strikingly, it shows that deciding whether two tensors generate the same uniform PEPS $|\mathbb{T}_\Gamma\rangle$ on arbitrary contraction graphs is **decidable** – in stark contrast to the problem when we restrict to uniform PEPS $|\mathbb{T}_{n,m}\rangle$ on periodic rectangular lattices. The undecidability of the latter is proved by relating it to the problem of deciding if a given set of tiles tiles a torus [SMG+20]. Our result implies that this problem becomes decidable if one allows for some arbitrary “surface” (contraction graph).

Given the practical and theoretical importance of canonical forms and fundamental theorems, we hope our results offer a useful new tool for the study and application of tensor networks. From a theory perspective, our results may be helpful in studying virtual symmetries of tensor networks, which are crucial in understanding topological order. From a practical perspective, it would be interesting to investigate if our canonical form can improve the numerical stability of variational optimization algorithms and other numerical methods [VHCV16], as it could be expected by the known close connection between gauge fixing and stability [LCB14a; PMV15]. Our results also imply that one can sample uniformly from all PEPS tensors in minimal canonical form in the same orbit. This has applications beyond quantum information, e.g., it allows to extend the technique of [PHM+22] for enhancing privacy in machine learning from MPS to PEPS. Finally, we note that our approach generalizes naturally to other tensor network types and

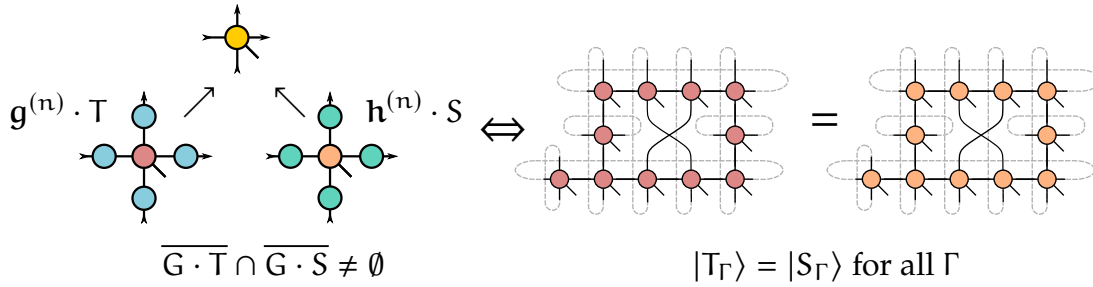


Figure 3.3.: *Fundamental theorem*: Two tensors S and T are gauge equivalent, meaning $\overline{G \cdot T} \cap \overline{G \cdot S} \neq \emptyset$ or that $\lim_{n \rightarrow \infty} \mathbf{g}^{(n)} \cdot T = \lim_{n \rightarrow \infty} \mathbf{h}^{(n)} \cdot S$ for certain $\mathbf{g}^{(n)}, \mathbf{h}^{(n)} \in G$ (equivalently, the two tensors have a common minimal canonical form), if and only if they contract to the same state on all contraction graphs.

gauge groups; it would be exciting to explore this in followup work. We discuss all these points further in Section 3.5.

3.1.2. Overview of methods: geometric invariant theory and geodesic convex optimization

On a high level, our approach is to start with the desired gauge symmetry and explore its natural consequences (rather than with a specific class of networks, such as PEPS on a torus). In our case this means starting with the action of the gauge group $G = \text{GL}(D_1) \times \cdots \times \text{GL}(D_m)$ on the vector space of PEPS tensors of a certain format, as above. To prove Results 1 and 2, we rely on the results of geometric invariant theory as developed in Chapter 2, in particular Mumford’s theorem (Theorem 2.3.7) and the Kempf–Ness theorem (Theorem 2.4.4). To prove Result 3, we instantiate the general framework of [BFG+19] (but give some improvements) and we relate the approximation guarantees provided by that framework to ℓ^2 -error (which is nontrivial).

So far, we have focused on geometry, but we now move to invariants to connect to tensor networks and sketch the proof of our fundamental theorem (Result 4). Mumford’s theorem (Theorem 2.3.7) implies that two tensors T, T' are gauge equivalent (meaning $\overline{G \cdot T} \cap \overline{G \cdot T'} \neq \emptyset$) if and only if $P(T) = P(T')$ for any G -invariant polynomial P . Now, for any contraction graph Γ , the tensor network state $|T_\Gamma\rangle$ is unchanged by gauge symmetries, and therefore its coefficients are G -invariant polynomials in T . We use constructive invariant theory to prove that, conversely, *any G -invariant polynomial can be obtained from coefficients of tensor network states*. A theorem by Derksen [Der00] allows bounding the size of Γ , which concludes the proof.

3.1.3. Notation

The baseline for the notation is as in Section 2.1. We write $\mathbf{y} = \text{argmin}\{f(x) : x \in X\}$ to denote that $\mathbf{y} \in X$ and $f(\mathbf{y}) = \min\{f(x) : x \in X\}$; in general this will not uniquely determine \mathbf{y} . In this chapter we write $\text{Mat}_{n \times n'}$ for the complex vector space of complex $n \times n'$ matrices, as opposed to $\mathbb{C}^{n \times n'}$. To be consistent with physicists’

3. The minimal canonical form of a tensor network

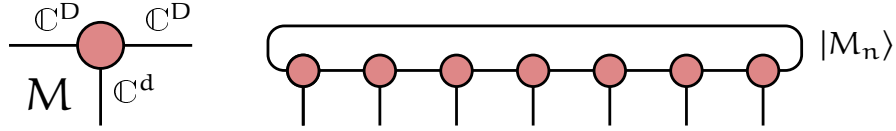


Figure 3.4.: *Matrix product state*: $M = (M^{(i)})_{i=1}^d \in \text{Mat}_{D \times D}^d$ gives rise to a state $|M_n\rangle$ for any system size n .

notation, we write $(\cdot)^\dagger$ instead of $(\cdot)^*$ for the adjoint (in this chapter only). We denote identity matrices by I and use subscripts to denote context when this increases clarity. We write $GL(n)$ for $GL(n, \mathbb{C})$ and $SL(n) = SL(n, \mathbb{C})$. We will use boldface for m -tuples of matrices, e.g., $\mathbf{X} = (X_1, \dots, X_m)$, but never for the d -tuples that make up uniform MPS or PEPS tensors. Finally, we denote by $\mathbb{C}[V]$ the algebra of polynomial functions on a vector space V .

3.2. Matrix product states

In this section, we discuss the setting of matrix product states (MPS). While MPS are very well-understood theoretically, it is instructive to revisit this setting from our new perspective and contrast our minimal canonical form to the known ones, which also enjoy excellent theoretical properties.

We start by defining uniform (or translation-invariant) MPS and briefly reviewing existing canonical forms in Section 3.2.1. We then introduce the minimal canonical form in Section 3.2.2. Finally, in Section 3.2.3 we also discuss the case of non-uniform MPS with open boundary conditions.

3.2.1. Gauge freedom and canonical forms for uniform MPS

We denote by $\text{Mat}_{D \times D}^d$ the vector space of d -tuples of $D \times D$ -matrices.

Definition 3.2.1 (Uniform MPS). For any matrix tuple $M = (M^{(i)})_{i=1}^d \in \text{Mat}_{D \times D}^d$ and system size $n \in \mathbb{N}$, we define the *uniform* (or *translation-invariant*) *matrix product state* (MPS) as the (not necessarily) quantum state $|M_n\rangle \in (\mathbb{C}^d)^{\otimes n}$ whose coefficients are given by

$$\langle i_1, \dots, i_n | M_n \rangle = \text{Tr} M^{(i_1)} \dots M^{(i_n)} \quad (\forall i_1, \dots, i_n \in [d]). \quad (3.2.1)$$

We refer to d as the *physical dimension* and D as the *bond dimension*.

We will interchangeably refer to M as a *matrix tuple* or as an *MPS tensor*. Indeed, it is often useful to think of M itself as a 3-tensor, or as an (unnormalized) quantum state $|M\rangle \in \mathbb{H}_1 \otimes \mathbb{H}_2 \otimes \mathbb{H}_{\text{phys}}$ on the tensor product of a physical Hilbert space $\mathbb{H}_{\text{phys}} = \mathbb{C}^d$ and two virtual Hilbert spaces $\mathbb{H}_1 = \mathbb{H}_2 = \mathbb{C}^D$, where \mathbb{H}_1 is the ‘left’ virtual Hilbert space and \mathbb{H}_2 is the ‘right’ virtual Hilbert space, see Fig. 3.4. Formally:

$$\langle a, b, i | M \rangle = \langle a | M^{(i)} | b \rangle.$$

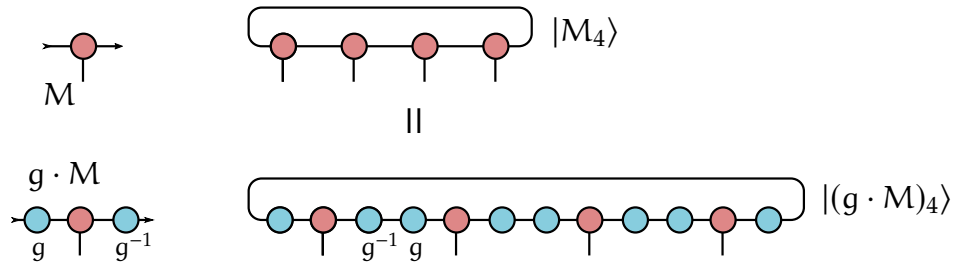


Figure 3.5.: *MPS gauge invariance*: Tensors related by a gauge transformation give rise to the same MPS.

We may then compute the reduced density matrices of $\rho = |M\rangle\langle M|$ on either of the two virtual Hilbert spaces:

$$\rho_1 = \sum_{i=1}^d M^{(i)}(M^{(i)})^\dagger \quad \text{and} \quad \rho_2 = \sum_{i=1}^d (M^{(i)})^\top \overline{M^{(i)}}. \quad (3.2.2)$$

An important property of MPS is that the states $|M_n\rangle$ are left invariant (for any n) if we conjugate each matrix $M^{(i)}$ in the tuple by the same invertible matrix. Formally:

Definition 3.2.2 (Gauge action). We define the *gauge action* of $g \in \text{GL}(D)$ on $M = (M^{(i)})_{i=1}^d$ by

$$g \cdot M := (gM^{(i)}g^{-1})_{i=1}^d.$$

If we think of M as a quantum state $|M\rangle$ in $\mathbb{H}_1 \otimes \mathbb{H}_2 \otimes \mathbb{H}_{\text{phys}}$, the gauge action can be written as

$$g \cdot |M\rangle := |g \cdot M\rangle = (g \otimes g^{-\top} \otimes I) |M\rangle.$$

Lemma 3.2.3 (Gauge symmetry). For every $M \in \text{Mat}_{D \times D}^d$, $g \in \text{GL}(D)$, and $n \in \mathbb{N}$, we have

$$|M_n\rangle = |(g \cdot M)_n\rangle.$$

This is shown in Fig. 3.5.

It is then a natural question to ask whether this is the only freedom in the tensor M to define the same state $|M_n\rangle$ for all n . The answer is *no*, as is well-known and illustrated by the following example:

Example 3.2.4. Let

$$M^{(0)} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad M^{(1)} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

and

$$\hat{M}^{(0)} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \hat{M}^{(1)} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

3. The minimal canonical form of a tensor network

Then both tensors define the same MPS, for any system size $n \in \mathbb{N}$, namely the *GHZ states*

$$|M_n\rangle = |\hat{M}_n\rangle = |0\rangle^{\otimes n} + |1\rangle^{\otimes n}.$$

However, there is no $g \in \text{GL}(D)$ so that $g \cdot M = \hat{M}$.

The underlying problem is that when the matrices $M^{(i)}$ in a tuple are all in upper triangular form (with respect to some basis), the off-diagonal terms are totally irrelevant for the final state $|M_n\rangle$. The standard way to deal with this is to *remove* such off-diagonal terms in a structured manner. Let us briefly sketch the procedure, but refer to [CPSV21] and [SPWC10] for details and nomenclature.

One starts looking for a minimal common invariant subspace of all $M^{(i)}$ and change $M^{(i)}$ by $PM^{(i)}P + QM^{(i)}Q$, with P being the orthogonal projector onto such a subspace and $Q = I - P$. It is not difficult to see that the new tensor defines the same original MPS. Now one proceeds similarly with $QM^{(i)}Q$ until one reaches a block diagonal form. The minimality of the subspaces guarantees that, in each of the diagonal blocks b , the corresponding tensor, say M_b , fulfills the property that the associated completely positive (CP) map \mathcal{E}_b given by $X_b \mapsto \sum_i M_b^{(i)} X_b (M_b^{(i)})^\dagger$ is *irreducible*. Normalizing so that the spectral radius of the map is 1, this implies that the eigenvalues of modulus 1 are all non degenerate and they are exactly the q -th roots of unity with a q dividing the size D_b of the matrices $M_b^{(i)}$. One can then distinguish two cases: $q = 1$, in which case the map \mathcal{E}_b is *primitive*, or $q > 1$, in which case one can “block” or group together q sites; then the resulting tensor in $\mathbb{H}_1 \otimes \mathbb{H}_2 \otimes \mathbb{H}_{\text{phys}}^{\otimes q}$ consists of block diagonal matrices whose associated CP maps are also primitive.

To make a long story short, starting with a matrix tuple M , after projecting and blocking following the above procedure, one obtains a new matrix tuple \tilde{M} such that each $\tilde{M}^{(i)}$ is block diagonal, $\tilde{M}^{(i)} = \oplus_b M_b^{(i)}$, and the CP maps \mathcal{E}_b are all primitive. It is now possible to act with a gauge $g \in \text{GL}(D)$, which can also be taken to be block-diagonal, $g = \oplus_b g_b$, so that one obtains in each block b of $\hat{M} := g \cdot \tilde{M}$ the *canonical* condition. That is, there exist constants $c_b \in \mathbb{R}_+$ such that

$$\sum_{i=1}^d (\hat{M}_b^{(i)})^\dagger \hat{M}_b^{(i)} = c_b I_{D_b} \quad (\forall b), \quad (3.2.3)$$

meaning that, after normalization, the maps $\mathcal{E}_b : X_b \mapsto \sum_i \hat{M}_b^{(i)} X_b (\hat{M}_b^{(i)})^\dagger$ are trace preserving completely positive (TPCP) maps, i.e., quantum channels. One could analogously have taken the *dual* condition

$$\sum_{i=1}^d \hat{M}_b^{(i)} (\hat{M}_b^{(i)})^\dagger = c_b I_{D_b} \quad (\forall b), \quad (3.2.4)$$

meaning the \mathcal{E}_b are completely positive unital (CPU) maps.

For generic matrix tuples M , the channel $X \mapsto \sum_i M^{(i)} X (M^{(i)})^\dagger$ is already primitive. In this case, M is called *normal* and one can obtain a left or right canonical form \hat{M} by acting with a suitable gauge group element: $\hat{M} = g \cdot M$ for some $g \in \text{GL}(D)$.

Definition 3.2.5 (Left and right canonical form). A matrix tuple (MPS tensor) is said to be in *left canonical form* if it is block diagonal, with each diagonal block a normal tensor fulfilling Eq. (3.2.3). The *right canonical form* is defined analogously by imposing the dual condition in Eq. (3.2.4).

The above procedure guarantees that, after discarding off-diagonal blocks and at the price of blocking, one can bring any MPS tensor into left or right canonical form. For instance, in Example 3.2.4 the tensor \hat{M} is block diagonal, its blocks are 1-dimensional and hence trivially primitive, and moreover $\hat{\rho}_1 = \hat{\rho}_2 = I_2$. Thus \hat{M} is both in left and right canonical form. For tensors in canonical form, (unitary) gauge symmetry is the only freedom for two tensors to generate the same MPS:

Theorem 3.2.6 (Fundamental theorem of MPS, [CPSV17; CPSV21]). Let M, N be both in left (or right) canonical form and $|M_n\rangle = |N_n\rangle$ for all $n \in \mathbb{N}$. Then there exists a unitary $u \in U(D)$ such that $u \cdot M = N$.

The name “*fundamental theorem*” stems from its numerous applications, and we refer for instance to [CPSV21] or [HV17] for an accounting of several of these.

3.2.2. The minimal canonical form for uniform MPS

We now define a new canonical form for uniform MPS. Its appeal is that it will naturally generalize to tensors with an arbitrary gauge symmetry and in particular to PEPS in higher dimensions, and that it can be analyzed using the powerful tools from geometric invariant theory.

Our starting point is the following simple but powerful observation: For a given matrix tuple $M \in \text{Mat}_{D \times D}^d$, we should not only consider gauge transformations $M \mapsto g \cdot M$ for some $g \in GL(D)$, but also limits of such. Indeed, suppose we have a sequence of gauge group elements $g_k \in GL(D)$ such that $g_k \cdot M$ converges to some \tilde{M} . Then, since the MPS $|M_n\rangle$ are continuous functions of the matrix tuple M , we still have

$$|\tilde{M}_n\rangle = \lim_{k \rightarrow \infty} |(g_k \cdot M)_n\rangle = |M_n\rangle \quad (\forall n \in \mathbb{N}).$$

In other words, all matrix tuples in the *orbit closure* $\overline{GL(D) \cdot M}$ determine the same MPS. This naturally leads to the following definition:

Definition 3.2.7 (Gauge equivalence). Let $M, N \in \text{Mat}_{D \times D}^d$ be two matrix tuples. We say that M and N are *gauge equivalent* if and only if $\overline{GL(D) \cdot M} \cap \overline{GL(D) \cdot N} \neq \emptyset$.

This is the natural notion of gauge equivalence for MPS tensors, since if M and N are gauge equivalent in the sense just defined then

$$|M_n\rangle = |N_n\rangle \quad (\forall n \in \mathbb{N}).$$

Indeed, it is the smallest equivalence relation generated by gauge transformations and taking limits. In particular, to define a canonical form we should naturally look at orbit closures, not just at orbits. How could we single out special elements in the orbit closure? The Kempf–Ness theorem (see Section 2.4) motivates the following definition:

3. The minimal canonical form of a tensor network

Definition 3.2.8 (Minimal canonical form of MPS). We say $M_{\min} \in \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$ is a *minimal canonical form* for a matrix tuple (MPS tensor) $M \in \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$ if it is an element of minimal norm in the orbit closure of the latter:

$$M_{\min} = \operatorname{argmin} \{ \|M'\| : M' \in \overline{\text{GL}(\mathbb{D}) \cdot M} \},$$

where we use the Euclidean norm of M (or $|M\rangle$), that is,

$$\|M\| = \sqrt{\langle M|M \rangle} = \left(\sum_{i=1}^d \operatorname{Tr} \left[(M^{(i)})^\dagger M^{(i)} \right] \right)^{1/2} = \left(\operatorname{Tr} \left[\sum_{i=1}^d (M^{(i)})^\dagger M^{(i)} \right] \right)^{1/2}.$$

We say $M \in \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$ is *in minimal canonical form* if it is a minimal canonical form for itself.

Note that any MPS tensor has a minimal canonical form – in contrast to the usual left or right canonical form of Definition 3.2.5, no explicit projecting and blocking is required.

Clearly, the minimal canonical form is a special case of the general notion of a minimum norm vector (Definition 2.4.1) for the action of $G = \text{GL}(\mathbb{D})$ on $V = \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$ (Definition 3.2.2). We can now use the general theory of geometric invariant theory to understand the basic properties of this canonical form and we will see the usefulness of the general results of Sections 2.3 and 2.4. First of all, while the minimal canonical form is not uniquely defined, it is uniquely defined up to unitary gauge transformations (the action of $K = \text{U}(\mathbb{D})$), and it precisely characterizes gauge equivalence (Definition 3.2.7):

Theorem 3.2.9 (Minimal canonical form). *Let $M, N \in \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$. Then the following are equivalent:*

- (i) M and N have a common minimal canonical form.
- (ii) If M_{\min}, N_{\min} are minimal canonical forms of M, N then $\text{U}(\mathbb{D}) \cdot M_{\min} = \text{U}(\mathbb{D}) \cdot N_{\min}$.
That is, minimal canonical forms of M and N are related by unitary gauge symmetries.
- (iii) M and N are gauge equivalent, i.e., $\overline{\text{GL}(\mathbb{D}) \cdot M} \cap \overline{\text{GL}(\mathbb{D}) \cdot N} \neq \emptyset$.

Proof. This is an immediate consequence of Theorems 2.3.7 and 2.4.4. □

The characterization of minimum norm vectors as critical vectors (Theorem 2.4.4) allows us to give an easy characterization for a matrix tuple to be in minimal canonical form. To see this, we compute the condition for a matrix tuple $M \in \text{Mat}_{\mathbb{D} \times \mathbb{D}}^d$ to be critical (Definition 2.4.3), i.e., we evaluate the moment map (Definition 2.5.1): For $X \in \text{Herm}(\mathbb{D}) = i\text{Lie}(\text{K})$, we have

$$\begin{aligned} \partial_{t=0} \|e^{tX} \cdot M\|^2 &= \partial_{t=0} \sum_{i=1}^d \operatorname{Tr} \left[(e^{tX} M^{(i)} e^{-tX})^\dagger e^{tX} M^{(i)} e^{-tX} \right] \\ &= \partial_{t=0} \sum_{i=1}^d \operatorname{Tr} \left[(M^{(i)})^\dagger e^{2tX} M^{(i)} e^{-2tX} \right] \\ &= 2 \operatorname{Tr} \left[X \left(\sum_{i=1}^d M^{(i)} (M^{(i)})^\dagger - (M^{(i)})^\dagger M^{(i)} \right) \right]. \end{aligned} \quad (3.2.5)$$

Thus we arrive at the following (illustrated in Fig. 3.6):

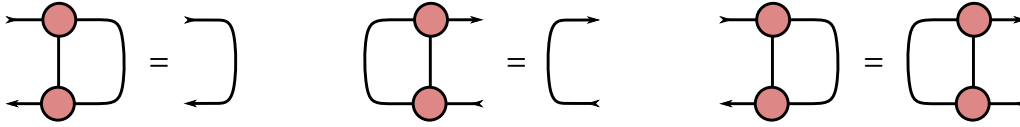


Figure 3.6.: MPS canonical forms: From left to right, the conditions for respectively right, left and minimal canonical forms for MPS.

Theorem 3.2.10 (Characterization). *Let $M \in \text{Mat}_{D \times D}^d$. Then M is in minimal canonical form if and only if $\|g \cdot M\| \geq \|M\|$ for all $g \in \text{GL}(D)$. This is the case if and only if*

$$\sum_{i=1}^d M^{(i)}(M^{(i)})^\dagger = \sum_{i=1}^d (M^{(i)})^\dagger M^{(i)}. \quad (3.2.6)$$

Equivalently, the reduced density matrices of $\rho = |M\rangle\langle M|$ on the virtual bonds are the same up to a transpose:

$$\rho_1 = \rho_2^\top. \quad (3.2.7)$$

Proof. Note that M is critical if and only if the derivative in Eq. (3.2.5) vanishes for all $X \in \text{Herm}(D)$. Thus both statements follow from Theorem 2.4.4. \square

Given a tensor M it is perhaps at first glance surprising that there always exist gauge transformations $g_k \in \text{GL}(D)$ such that $\lim_{k \rightarrow \infty} g_k \cdot M$ satisfies the condition in Eqs. (3.2.6) and (3.2.7) – yet as we just saw this follows readily from geometric invariant theory. We also note that Theorem 3.2.10 also shows that the minimal canonical form for MPS will in general *not* coincide with the usual left or right canonical form (Definition 3.2.5); there appears to be no obvious way to convert one into the other. In Section 3.4 we give a simple iterative algorithm that computes the minimal canonical form to arbitrary precision.

To get more intuition about the definition and the relevance of the orbit closure, we revisit Example 3.2.4.

Example 3.2.11. In Example 3.2.4 we saw that the following matrix tuples $M, \hat{M} \in \text{Mat}_{2 \times 2}^2$ both define the GHZ states:

$$M^{(0)} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \quad M^{(1)} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \hat{M}^{(0)} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \hat{M}^{(1)} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

Theorem 3.2.10 shows that \hat{M} is already in minimal canonical form, while M is not. Indeed, while $\hat{\rho}_1 = \hat{\rho}_2^\top = I_2$ for $\hat{\rho} = |\hat{M}\rangle\langle \hat{M}|$, the reduced states of $\rho = |M\rangle\langle M|$ satisfy

$$\rho_1 = M^{(0)}(M^{(0)})^\dagger + M^{(1)}(M^{(1)})^\dagger = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix},$$

$$\rho_2^\top = (M^{(0)})^\dagger M^{(0)} + (M^{(1)})^\dagger M^{(1)} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

3. The minimal canonical form of a tensor network

Moreover, in this example it is easy to see that there does *not* exist a $g \in \text{GL}(2)$ such that $g \cdot M$ is in minimal canonical form. However, if we let

$$g_\varepsilon = \begin{bmatrix} \varepsilon & 0 \\ 0 & 1 \end{bmatrix}$$

then we may verify that

$$\begin{aligned} g_\varepsilon M^{(0)} g_\varepsilon^{-1} &= \begin{bmatrix} \varepsilon & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \varepsilon^{-1} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & \varepsilon \\ 0 & 0 \end{bmatrix} \\ g_\varepsilon M^{(1)} g_\varepsilon^{-1} &= \begin{bmatrix} \varepsilon & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \varepsilon^{-1} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & \varepsilon \\ 0 & 1 \end{bmatrix} \end{aligned}$$

so as we let $\varepsilon \rightarrow 0$ we see that $g_\varepsilon \cdot M \rightarrow \hat{M}$, which as just discussed is in minimal canonical form.

Example 3.2.12. An amusing special case is $d = 1$, so we have a single matrix $M \in \text{Mat}_{D \times D}$. The minimal canonical form is given by the diagonal matrix with the same eigenvalues as M (repeated according to their algebraic multiplicity). Indeed, there are matrices g_ε such that $g_\varepsilon \cdot M = g_\varepsilon M g_\varepsilon^{-1}$ is in Jordan normal form, but with ε instead of 1 as the offdiagonal entries. Letting $\varepsilon \rightarrow 0$ we obtain the desired diagonal matrix.

From Examples 3.2.11 and 3.2.12 it is clear that, by virtue of considering the orbit closure, the minimal canonical form *automatically* sets off-diagonal blocks to zero, which is an additional step which needs to be manually taken in the usual approach to canonical forms for MPS (see Section 3.2.1). There, as already commented in Section 3.2.1, it may also be necessary to block together multiple sites. The geometric invariant theory approach makes these steps redundant.³

We will now prove a fundamental theorem for MPS where this will become explicit. Before stating the result, we state the ingredient that will be used to prove it. In invariant theory, the action of the gauge group on MPS tensors (Definition 3.2.2) is known as the *simultaneous conjugation action* of $\text{GL}(D)$ on matrix tuples in $\text{Mat}_{D \times D}^d$. There, it is known that the ring of invariant polynomials is generated precisely by the coefficients (3.2.1) of the corresponding matrix product states for system size $1 \leq n \leq D^2$, as stated in the following theorem:

Theorem 3.2.13 (Procesi-Razmyslov-Formanek [Pro76; Raz74; For86; DP17]). *The invariant ring for the simultaneous conjugation action, i.e., $\mathbb{C}[\text{Mat}_{D \times D}^d]^{\text{GL}(D)}$, is generated by the invariant polynomials $P_{\mathbf{i}}$, where*

$$P_{\mathbf{i}}(M) = \langle i_1, \dots, i_n | M_n \rangle = \text{Tr } M^{(i_1)} \dots M^{(i_n)},$$

for all $\mathbf{i} = (i_1, \dots, i_n) \in [d]^n$ and $n \in \mathbb{N}$. Moreover, it suffices to restrict to $n \in [D^2]$.

Thus, geometric invariant theory implies that gauge equivalence of the tensors (which by Theorem 3.2.9 is captured by the minimal canonical form) is precisely equivalent to equality of the corresponding matrix product states! We summarize this in the following fundamental theorem for MPS (note that it works in full generality, without the need to block sites or remove off-diagonal terms):

³As a side remark, there is actually no need to block in the usual canonical form for MPS. This is a consequence of Theorem 16 in [DCSP17], together with the overlooked observation that the matrix Z appearing there can be absorbed in another gauge transformation.

Theorem 3.2.14 (Fundamental theorem for MPS). *Let $M, N \in \text{Mat}_{D \times D}^d$. Then the following are equivalent:*

- (i) M and N are gauge equivalent, i.e., $\overline{\text{GL}(D) \cdot M} \cap \overline{\text{GL}(D) \cdot N} \neq \emptyset$.
- (ii) $|M_n\rangle = |N_n\rangle$ for all $n \in \mathbb{N}$.
- (iii) $|M_n\rangle = |N_n\rangle$ for $n = 1, \dots, D^2$.

Proof. This follows from Theorems 2.3.7 and 3.2.13. □

Remark 3.2.15. *It is also known that the invariant ring is not generated when restricting to $n \leq D^2/8$ [For86]. However, while a system of generators of the invariant ring always suffices to separate orbit closures, this is in fact not necessary. Theorem 1.14 in [DM20a] shows that the third condition in Theorem 3.2.14 can be improved almost quadratically to:*

$$3'. |M_n\rangle = |N_n\rangle \text{ for } n = 1, \dots, 4D \log_2 D + 12D - 4,$$

and it has been conjectured that $n = O(D)$ suffices [Shi19]. Example 3.2.17 shows that this is essentially tight.

Example 3.2.16. In Example 3.2.4 we saw two matrix tuples $M, \hat{M} \in \text{Mat}_{2 \times 2}^2$ that defined the GHZ states, for all system sizes. By our fundamental theorem, Theorem 3.2.14, this implies that they are gauge equivalent, meaning that

$$\overline{\text{GL}(D) \cdot M} \cap \overline{\text{GL}(D) \cdot \hat{M}} \neq \emptyset.$$

Now, in Example 3.2.11 we saw that \hat{M} is already in minimal canonical form. By the Kempf–Ness theorem (Theorem 2.4.4) this means that the orbit of \hat{M} is already closed. It follows that

$$\hat{M} \in \overline{\text{GL}(D) \cdot M},$$

which is in exact agreement with what we saw in Example 3.2.11.

Example 3.2.17. We also revisit Example 3.2.12, the case of a single matrix. For $M, N \in \text{Mat}_{D \times D}$, the equality of quantum states means that $\text{Tr } M^n = \text{Tr } N^n$ for all n , which is the case if and only if M, N have the same characteristic polynomial and hence the same eigenvalues with the same algebraic multiplicities – in agreement with the discussion in Example 3.2.12. Thus we see that in this special case it suffices to have equality for all $n = 1, \dots, D$. This is also necessary, since, e.g., for M a $D \times D$ -permutation matrix representing a D -cycle we have $\text{Tr } M^n = 0$ for $1 \leq n < D$.

Together, Theorems 3.2.9 and 3.2.14 show that if M, N are two matrix tuples in minimal canonical form that give rise to the same quantum states, then M and N are related by a *unitary* gauge symmetry. As a consequence, we can lift unitary symmetries to the virtual level. Again, we do not need to make any assumptions about the tensor M .

Corollary 3.2.18 (Lifting symmetries). *Suppose that $M, N \in \text{Mat}_{D \times D}^d$ are in minimal canonical form and $u \in U(d)$ is a unitary such that $u^{\otimes n} |M_n\rangle = |N_n\rangle$ for all $n \in \mathbb{N}$. Then there exists a unitary $U \in U(D)$ such that $(I \otimes I \otimes u) |M\rangle = (U \otimes \hat{U} \otimes I) |N\rangle$.*

3. The minimal canonical form of a tensor network

In other words, the action of u on the physical degrees of M is implemented by the gauge action of U on N .

Proof. Let $M' \in \text{Mat}_{D \times D}^d$ be the matrix tuple defined by

$$|M'\rangle := (I \otimes I \otimes u) |M\rangle.$$

Then M' is also in minimal canonical form, since u is unitary and hence we have $\|g \cdot M\| = \|g \cdot M'\|$ for all $g \in \text{GL}(D)$. Moreover, by construction it holds that

$$|M'_n\rangle = u^{\otimes n} |M_n\rangle = |N_n\rangle$$

for all $n \in \mathbb{N}$. Thus Theorem 3.2.14 shows that M' and N are gauge equivalent, and it follows from Theorem 3.2.9 that there exists a unitary gauge transformation $U \in \text{GL}(D)$ such that $U \cdot N = M'$. \square

We note that U need not be unique; for instance, M itself may have a stabilizer, i.e., there may exist $U \in \text{U}(D)$ such that $U \cdot M = M$. Indeed, this is exactly the case in which the MPS given by M has a global on-site symmetry, for which Corollary 3.2.18 reproduces, for the minimal canonical form, the known local characterization of symmetries on MPS [CPSV21] usually obtained via the left or right canonical form and Theorem 3.2.6.

Such characterization is the key step in the classification of symmetry protected topological phases done in [CGW11; PBTO12; SPC11]. The connection is as follows. If a system is invariant under the action of an onsite (global) symmetry group u_g , one gets $u_g^{\otimes n} |\Psi_n\rangle = |\Psi_n\rangle$ for its ground state $|\Psi_n\rangle$ (global phases do not play a relevant role here). Since $|\Psi_n\rangle$ is known to be very well approximated by MPS one may want to solve equation $u_g^{\otimes n} |M_n\rangle = |M_n\rangle$ for the MPS generated by some tensor M . By Corollary 3.2.18, this is characterized by the existence of $U_g \in \text{U}(D)$ such that $(I \otimes I \otimes u_g) |M\rangle = (U_g \otimes \bar{U}_g \otimes I) |M\rangle$. It is not difficult to see that U_g must be a projective representation of the symmetry group. The classification of SPT phases is given then by all non-equivalent projective representations, which is precisely described by the second cohomology group of the group cohomology of the symmetry group. The general validity of this approach has been recently established by the groundbreaking results of Ogata [Oga20].

The idea that the relevant topological content of a system lies in its boundary has also given rise to the study of a bulk-boundary correspondence, usually known in this context as “entanglement spectra” or “entanglement Hamiltonian” [CPSV11], in which one upgrades the boundary to a physical system and looks for a dictionary between bulk and boundary properties. This is precisely the reason that tensor networks have become rather popular in the context of AdS-CFT holography in quantum gravity. For this program it is rather crucial that the boundary representations of the physical on-site symmetries are indeed given themselves by unitary representations, which is precisely what Corollary 3.2.18 guarantees for the MPS case.

Remark 3.2.19. *As commented in Section 3.2.1, a MPS state can also be interpreted as a CP map on the virtual Hilbert spaces, where $M \in \text{Mat}_{D \times D}^d$ is interpreted such that the $M^{(i)}$ are Kraus operators of a CP map \mathcal{E} , usually called the transfer operator. Equivalently, the reduced state ρ_{12} of the quantum state $\rho = |M\rangle \langle M|$ on both virtual*

Hilbert spaces is the Choi operator of \mathcal{E} . As explained above Definition 3.2.5, the left and right canonical form conditions are equivalent to \mathcal{E} either being completely positive trace-preserving (CPTP) or unital (CPU). This perspective is particularly useful when dealing with contractions of large or infinite uniform MPS (the thermodynamic limit).

What is the interpretation of the minimal canonical form in this perspective? It is not hard to see that a mixed quantum state ρ_{12} with conjugate marginals (i.e., $\rho_1 = \rho_2^\top$) that are full-rank contains exactly the same data as a CPTP map Φ along with a full-rank invariant density operator Ω (i.e., $\Phi(\Omega) = \Omega$). The isomorphism $\rho_{12} \mapsto (\Phi, \Omega)$ is defined by defining $\Phi = \Phi_{1 \rightarrow 2}$ as the CPTP map with Choi operator $\rho_1^{-1/2} \rho_{12} \rho_1^{-1/2}$ and $\Omega = \rho_2 = \rho_1^\top$. If the marginals do not have full rank we can restrict to its support. By duality, this is in turn the same as a CP unital map ϕ along with a faithful invariant state ω in the algebraic sense: We have an isomorphism $(\Phi, \Omega) \mapsto (\phi, \omega)$, defined by taking $\phi = \Phi^\dagger$ and $\omega(X) = \text{Tr } \Omega X$. At this point we do not see a natural interpretation of these conditions for MPS contractions in the thermodynamic limit.

3.2.3. Canonical forms for MPS with open boundary conditions

We will now consider open boundary conditions. We use the invariant theory framework to define canonical forms, which in this case are closely related to well-known canonical forms. Then it is natural to fix the system size n , and to consider the non-uniform setting. Let

$$V = \bigoplus_{k=0}^{n-1} \text{Mat}_{D_k \times D_{k+1}}^d$$

where $D_0 = D_n = 1$. As usual, d is the physical dimension and the D_k are the bond dimensions (which may vary per bond). Let $\mathbf{M} = (M_0, \dots, M_{n-1}) \in V$, then the associated MPS state $|M\rangle$ (note that now we have a fixed system size) is defined by

$$\langle i_0 \dots i_{n-1} | M \rangle = M_0^{(i_0)} M_1^{(i_1)} \dots M_{n-1}^{(i_{n-1})}.$$

We let $G = \text{GL}(D_1) \times \dots \times \text{GL}(D_{n-1})$ act on V by gauge transformations. To define this action, let $\mathbf{g} = (g_1, \dots, g_{n-1}) \in G$ and $\mathbf{M} = (M_0, \dots, M_{n-1}) \in V$. Then the action is given by

$$\mathbf{g} \cdot \mathbf{M} = ((1, g_1) \cdot M_0, (g_1, g_2) \cdot M_1, \dots, (g_{n-2}, g_{n-1}) \cdot M_{n-2}, (g_{n-1}, 1) \cdot M_{n-1}).$$

where for $M_i = (M_i^{(j)})_{j=1}^d$

$$(g_i, g_{i+1}) \cdot M_i := (g_i M_i^{(j)} g_{i+1}^{-1})_{j=1}^d.$$

It is clear that the resulting MPS state is invariant under the action of G . For every ‘bond cut’ $k \in \{1, \dots, n-1\}$, we let

$$W_k := \text{Mat}_{d^k \times D_k} \oplus \text{Mat}_{D_k \times d^{n-k}}$$

and we define a G -action on W_k by

$$\mathbf{g} \cdot (w_{\text{left}}, w_{\text{right}}) = (w_{\text{left}} g_k^{-1}, g_k w_{\text{right}}).$$

3. The minimal canonical form of a tensor network

Then we have a map $\iota_k: V \rightarrow W_k$, which maps the vector of MPS tensors \mathbf{M} to a pair of 'half-chain contractions' $(M_{k,\text{left}}, M_{k,\text{right}})$

$$\langle i_0 \dots i_{k-1} | M_{k,\text{left}} = M_0^{(i_0)} M_1^{(i_1)} \dots M_{k-1}^{(i_{k-1})}, M_{k,\text{right}} | i_k \dots i_{n-1} \rangle = M_k^{(i_k)} \dots M_{n-1}^{(i_{n-1})}.$$

This map is clearly G -equivariant. We can patch the maps ι_k together to obtain a G -equivariant polynomial map

$$\iota: V \rightarrow W := \bigoplus_{k=1}^{n-1} W_k.$$

We can think of $M_{k,\text{left}}$ and $M_{k,\text{right}}$ as the states where we have contracted all the bonds except the k -th. In this perspective the reduced density matrices on the left and right copies of \mathbb{C}^{D_k} are given by

$$\begin{aligned} \rho_{k,\text{left}} &= \sum_{\mathbf{i}} (M_{k-1}^{(i_{k-1})})^\dagger \dots (M_0^{(i_0)})^\dagger M_0^{(i_0)} \dots M_{k-1}^{(i_{k-1})} = M_{k,\text{left}}^\dagger M_{k,\text{left}} \\ \rho_{k,\text{right}}^\top &= \sum_{\mathbf{i}} M_k^{(i_k)} \dots M_{n-1}^{(i_{n-1})} (M_{n-1}^{(i_{n-1})})^\dagger \dots (M_k^{(i_k)})^\dagger = M_{k,\text{right}} M_{k,\text{right}}^\dagger \end{aligned}$$

We claim that norm minimization in the image of ι leads to a canonical form where $\rho_{k,\text{left}} = \rho_{k,\text{right}}^\top$, which we call the *minimal canonical form* for non-uniform MPS:

Definition 3.2.20. Let $\mathbf{M} \in V$. Then \mathbf{M}_{min} is a *minimal canonical form* for \mathbf{M} if $\iota(\mathbf{M}_{\text{min}})$ is an element of minimal norm with respect to the orbit closure $\overline{G \cdot \mathbf{M}}$, i.e.,

$$\mathbf{M}_{\text{min}} = \operatorname{argmin} \{ \|\iota(\mathbf{M}')\| : \mathbf{M}' \in \overline{GL(D) \cdot \mathbf{M}} \}.$$

The norm we are considering here is again the Euclidean one. Note also that $\mathbf{g} \cdot \iota_k(\mathbf{M})$ only depends on g_k . Therefore, we may also write $g_k \cdot \iota_k(\mathbf{M})$. In minimizing $\|\mathbf{g} \cdot \iota(\mathbf{M})\|$ we may minimize each $\|g_k \cdot \iota_k(\mathbf{M})\|$ separately. By the same general theory as applied in Section 3.2.2 we deduce that the canonical form exists and is unique up to conjugation by unitary elements in G . Moreover, as in Theorem 3.2.10 we may set an appropriate derivative equal to zero to find a condition for when \mathbf{M} is in minimal canonical form.

Letting $g_k(t) = e^{tX_k}$ for $X_k \in \operatorname{Herm}(D_k)$ we see that

$$\begin{aligned} &\|g_k(t) \cdot \iota_k(\mathbf{M})\|^2 \\ &= \operatorname{Tr} \left[(g_k(t)^{-1})^\dagger M_{k,\text{left}}^\dagger M_{k,\text{left}} g_k(t)^{-1} + g_k(t) M_{k,\text{right}} M_{k,\text{right}}^\dagger g_k(t)^\dagger \right] \\ &= \operatorname{Tr} \left[e^{-2tX_k} M_{k,\text{left}}^\dagger M_{k,\text{left}} + M_{k,\text{right}} M_{k,\text{right}}^\dagger e^{2tX_k} \right] \end{aligned}$$

and hence, denoting by $\mathbf{g}(t) = (g_1(t), \dots, g_{n-1}(t))$ we have

$$\begin{aligned} \partial_{t=0} \|\mathbf{g}(t) \cdot \iota(\mathbf{M})\|^2 &= \partial_{t=0} \sum_{k=1}^{n-1} \|g_k(t) \cdot \iota_k(\mathbf{M})\|^2 \\ &= 2 \sum_{k=1}^{n-1} \operatorname{Tr} \left[X_k \left(M_{k,\text{right}} M_{k,\text{right}}^\dagger - M_{k,\text{left}}^\dagger M_{k,\text{left}} \right) \right]. \end{aligned}$$

Setting this equal to zero is equivalent to $\rho_{k,\text{left}} = \rho_{k,\text{right}}^\top$ for all k .

We may explicitly perform the minimization; it is closely related to Vidal's canonical form [Vid03]. We can perform a singular value decomposition

$$M_{k,\text{left}} = V_1 \Sigma_1 U_1$$

where $V_1 \in \text{Mat}_{d^k \times D_k}$ is an isometry, $\Sigma_1 \in \text{Mat}_{D_k \times D_k}$ is diagonal with nonnegative entries and $U_1 \in \text{Mat}_{D_k \times D_k}$ is unitary. Next, we perform a singular value decomposition on $\Sigma_1 U_1 M_{k,\text{right}}$ so

$$\Sigma_1 U_1 M_{k,\text{right}} = U_2 \Sigma_2 V_2$$

where $V_2 \in \text{Mat}_{D_k \times d^{n-k}}$ is an isometry, $\Sigma_2 \in \text{Mat}_{D_k \times D_k}$ is diagonal with nonnegative entries and $U_2 \in \text{Mat}_{D_k \times D_k}$ is unitary. Let Π_i be the projection onto $\ker(\Sigma_i)$ and let $\tilde{\Sigma}_i = \Sigma_i + \Pi_i$. Then let

$$g_k = \sqrt{\tilde{\Sigma}_2^{-1}} U_2^\dagger \tilde{\Sigma}_1 U_1$$

and we let $\tilde{M}_{k,\text{left}} = M_{k,\text{left}} g^{-1}$ and $\tilde{M}_{k,\text{right}} = g M_{k,\text{right}}$. Then we may verify that the associated reduced density matrices are

$$\begin{aligned} \rho_{k,\text{left}} &= \tilde{M}_{k,\text{left}}^\dagger \tilde{M}_{k,\text{left}} \\ &= \sqrt{\tilde{\Sigma}_2} U_2^\dagger \tilde{\Sigma}_1^{-1} U_1 M_{k,\text{left}}^\dagger M_{k,\text{left}} U_1^\dagger \tilde{\Sigma}_1^{-1} U_2 \sqrt{\tilde{\Sigma}_2} \\ &= \sqrt{\tilde{\Sigma}_2} U_2^\dagger \tilde{\Sigma}_1^{-1} U_1 U_1^\dagger \Sigma_1 V_1^\dagger V_1 \Sigma_1 U_1 U_1^\dagger \tilde{\Sigma}_1^{-1} U_2 \sqrt{\tilde{\Sigma}_2} \\ &= \Sigma_2 \end{aligned}$$

and

$$\begin{aligned} \rho_{k,\text{right}}^\top &= \tilde{M}_{k,\text{right}} \tilde{M}_{k,\text{right}}^\dagger \\ &= \sqrt{\tilde{\Sigma}_2^{-1}} U_2^\dagger \tilde{\Sigma}_1 U_1 M_{k,\text{right}} M_{k,\text{right}}^\dagger U_1^\dagger \tilde{\Sigma}_1 U_2 \sqrt{\tilde{\Sigma}_2^{-1}} \\ &= \sqrt{\tilde{\Sigma}_2^{-1}} U_2^\dagger U_2 \Sigma_2 U_2 U_2^\dagger \Sigma_2 U_2^\dagger U_2 \sqrt{\tilde{\Sigma}_2^{-1}} \\ &= \Sigma_2. \end{aligned}$$

Therefore, defining g_k in this fashion for each k gives $g \cdot M$ in minimal canonical form. In this case it is not necessary to go to the closure to obtain the canonical form.

This canonical form coincides with the one of Vidal [Vid03], usually written in the form

$$\sum_{i_0 \dots i_{n-1}} \Gamma_0^{(i_0)} \Lambda_1 \Gamma_1^{(i_1)} \Lambda_2 \cdots \Lambda_{n-1} \Gamma_{n-1}^{(i_{n-1})} |i_0, i_{n-1}\rangle \quad (3.2.8)$$

if one identifies $M_k^{(i_k)}$ with $\sqrt{\Lambda_k} \Gamma_k^{(i_k)} \sqrt{\Lambda_{k+1}}$. The reason is that, by the properties of Vidal's canonical form [Vid03; Sch11], such choice fulfills the algebraic characterization of the minimal canonical form given by $\rho_{k,\text{left}} = \rho_{k,\text{right}}^\top$ for all k . Since the positive diagonal matrices Λ_k correspond to the Schmidt coefficients of the

3. The minimal canonical form of a tensor network

bipartition of the system in the cut $[0 : k - 1], [k - 1]$, the minimal canonical form can be understood in this case as an even distribution of those weights. This particular distribution of weights has also appeared extensively in the standard MPS literature [OV08].

There are also left and right canonical forms [Sch11]. These fit in the same framework, which we will now show for the left canonical form (with the right canonical form being completely analogous). Let V be as before, but now we consider the action of $G = \text{SL}(D_1) \times \cdots \times \text{SL}(D_{n-1})$. We let $W_k = \text{Mat}_{d^k \times D_k}$ (which is only the left half chain) and we let $\iota_k : V \rightarrow W_k$ be given by $M_k = \iota_k(\mathbf{M})$

$$\langle i_0 \dots i_{k-1} | M_k = M_0^{(i_0)} M_1^{(i_1)} \dots M_{k-1}^{(i_{k-1})}$$

(so this is what previously was $M_{k,\text{left}}$). The group action is given by the $M_k \mapsto M_k g_k^{-1}$. We similarly define

$$\iota : V \rightarrow W := \bigoplus_{k=1}^{n-1} W_k.$$

Computing the gradient as before, but now restricting to *traceless* X (as we are optimizing over $\text{SL}(D_k)$) we find that at the minimum of the norm $\|\mathbf{g} \cdot \iota(\mathbf{M})\|$ the reduced density matrix $\rho_{k,\text{left}}$ must be proportional to the identity for all k . Again, we can explicitly realize the minimum, without going to the closure. To this end we perform a singular value decomposition $M_k = V \Sigma U$. Let Π be the projection onto $\ker(\Sigma)$ and let $\tilde{\Sigma} = \Sigma + \Pi$. Then taking $g_k = \det(\tilde{\Sigma} U)^{-1/D_k} \tilde{\Sigma} U \in \text{SL}(D_k)$ yields a uniform reduced density matrix $\rho_{k,\text{left}}$.

3.3. Projected entangled pair states

In this section we start by defining projected entangled pair states (PEPS), in particular uniform PEPS. In Section 3.3.2 we introduce the minimal canonical form for PEPS. We will see that by closely analogous arguments to the MPS case we may establish its basic properties. In Section 3.3.4 we relate to two-dimensional tilings and explain how our results are compatible with earlier no-go results for the existence of canonical forms for PEPS. In Section 3.3.5 we study in more detail the role of the orbit closure and show that in many cases of interest the orbit is closed.

3.3.1. Definition of uniform PEPS

We will now define a generalization of MPS, known as Projected Entangled Pair States (PEPS). We start by defining a rather general version, and then specialize to cases of interest. As input we require a graph $\Gamma = (V, E)$ and dimensions $(D_e)_{e \in E}$ (the bond dimensions) and $(d_v)_{v \in V}$ (the physical dimensions). Let $E(v)$ denote the set of edges incident to $v \in V$. Then we let $\mathbb{H}_v := \mathbb{C}^{d_v}$ and for each $e \in E(v)$ we let $\mathbb{H}_{v,e} := \mathbb{C}^{D_e}$. The PEPS will now be constructed from a collection of tensors $(T^{[v]})_{v \in V}$ where

$$T^{[v]} \in \left(\bigotimes_{e \in E(v)} \mathbb{H}_{v,e} \right) \otimes \mathbb{H}_v.$$

The resulting PEPS is a state on $\bigotimes_{v \in V} \mathbb{H}_v$ and is constructed by ‘contracting along the edges’. If $e = (v, w)$ is an edge incident to v and w , then the contraction map $\delta_e : \mathbb{H}_{v,e} \otimes \mathbb{H}_{w,e} \rightarrow \mathbb{C}$ along e is defined by

$$|ij\rangle \mapsto \delta_{i,j}$$

and extending by linearity. We may apply these maps along each of the edges in E and this yields a state $|\Gamma\rangle$ on $\bigotimes_{v \in V} \mathbb{H}_v$.

A clean way of writing this contraction operation (and also explaining the nomenclature projected entangled pair states) is by the identity

$$|\Gamma\rangle = \left(\bigotimes_{e=(v,w) \in E} \left(\sum_{i=0}^{D_e-1} \langle ii| \right) \otimes I_V \right) \bigotimes_{v \in V} T^{[v]}.$$

where I_V is the identity operator on $\bigotimes_{v \in V} \mathbb{H}_v$.

We will now specialize to the case of *uniform PEPS*. In this case we place the same tensor at each vertex. It is natural to contract the tensors placed on periodic grids in m spatial dimensions, but we will see that other graphs are also relevant. We denote the physical dimension by d and the associated physical Hilbert space by $\mathbb{H}_{\text{phys}} = \mathbb{C}^d$, and there are m relevant bond dimensions in the different directions, which we will denote by D_k for $k \in [m]$. For each direction $k \in [m]$ we have two Hilbert spaces $\mathbb{H}_{k,1} = \mathbb{C}^{D_k}$ and $\mathbb{H}_{k,2} = \mathbb{C}^{D_k}$. Similar to the MPS case, we may interpret the PEPS tensor T either as a tensor

$$|T\rangle \in \left(\bigotimes_{k=1}^m \mathbb{H}_{k,1} \otimes \mathbb{H}_{k,2} \right) \otimes \mathbb{H}_{\text{phys}} \quad (3.3.1)$$

or as a matrix tuple

$$T = (T^{(i)})_{i=1}^d, \quad T^{(i)} \in \bigotimes_{j=1}^m \text{Mat}_{D_j \times D_j} \quad (3.3.2)$$

and we will generally identify this space of matrix tuples as $\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$. Typically, one constructs corresponding quantum states by placing copies of the tensor on a grid and contracting along the bond dimensions, see Fig. 3.7.

Definition 3.3.1 (Uniform PEPS on a grid). For any matrix tuple $T = (T^{(i)})_{i=1}^d \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ and system sizes $n_1, \dots, n_m \in \mathbb{N}$, we define the *uniform* (or *translation-invariant*) *projected entangled pair state* (PEPS) as the (not necessarily) quantum state $|T_{n_1, \dots, n_m}\rangle \in (\mathbb{C}^d)^{\otimes n}$, where $n = n_1 \dots n_m$ and which is given by contracting n copies of T on an $n_1 \times \dots \times n_m$ periodic grid.

We would like to allow a broader class of uniform PEPS, where one may use in principle any possible contraction graph. In such a contraction graph we only demand that the directions are matched up, in the sense that we always contract $\mathbb{H}_{k,1}$ with $\mathbb{H}_{k,2}$. A natural way to express such contractions is as follows. Suppose that we have n vertices, with at each vertex a copy of T , and we are given a contraction graph. We will define permutations $\pi_k \in S_n$ for each direction $k \in [m]$.

3. The minimal canonical form of a tensor network

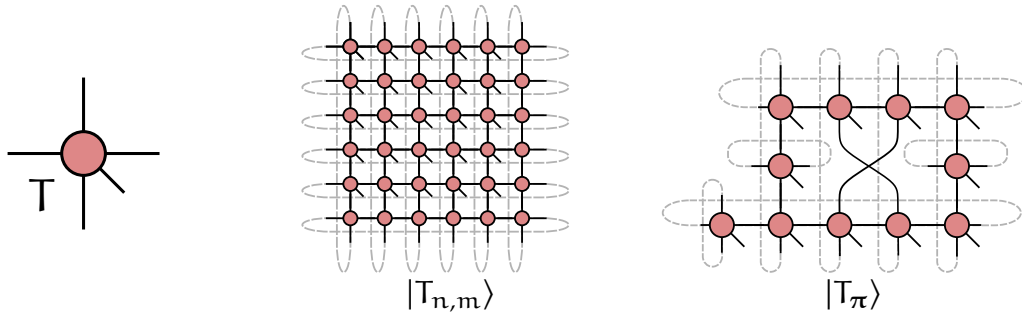


Figure 3.7.: *Projected entangled pair states*: Given a tensor T , here in two spatial dimensions, we may contract on a $n_1 \times n_2$ grid to obtain $|T_{n_1, n_2}\rangle$ or using arbitrary permutations $\pi = (\pi_1, \pi_2)$ to get $|T_{\pi}\rangle$.

Suppose that in direction k $\alpha, \beta \in [n]$ are such that the Hilbert space $\mathbb{H}_{k,2}$ of the α -th copy of T is contracted with the Hilbert space $\mathbb{H}_{k,1}$ of the β -th copy of T , then we let π_k map α to β . Each contraction map (and ordering of the vertices) then uniquely determines permutations $\pi_k \in S_n$. As permutations $\pi = (\pi_1, \dots, \pi_m)$ completely determine the contraction of the n copies of T to a quantum state on $\mathbb{H}_{\text{phys}}^{\otimes n} = (\mathbb{C}^d)^{\otimes n}$ we denote this state by $|T_{\pi}\rangle$. For $k \in [m]$ let R_{π_k} be the operator on $(\mathbb{C}^{D_k})^{\otimes n}$ permuting the n tensor factors.

Definition 3.3.2 (Uniform PEPS on arbitrary contraction graphs). For any matrix tuple $T = (T^{(i)})_{i=1}^d \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$, system size n and for $\pi = (\pi_1, \dots, \pi_m) \in S_n^m$ we define the associated *uniform projected entangled pair state (PEPS)* as the (not necessarily) quantum state $|T_{\pi}\rangle \in (\mathbb{C}^d)^{\otimes n}$ which has coefficients defined by

$$\langle i_1, \dots, i_n | T_{\pi} \rangle = \text{Tr} \left[(R_{\pi_1} \otimes \dots \otimes R_{\pi_m}) T^{(i_1)} \otimes \dots \otimes T^{(i_n)} \right] \quad \mathbf{i} = (i_1, \dots, i_n) \in [d]^n.$$

We may use the coefficients of the contracted state $|T_{\pi}\rangle$ to define functions $P_{\pi, \mathbf{i}} \in \mathbb{C}[\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d]$ as

$$P_{\pi, \mathbf{i}}(T) = \langle i_1, \dots, i_n | T_{\pi} \rangle. \quad (3.3.3)$$

For $m = 1$ we get back the usual notion of MPS. Note that in this case, if we assume the contraction graph to be connected, there is a unique way to contract the tensors, corresponding to any full cycle in S_n . Indeed, for $T \in \text{Mat}_{D \times D}^d$ and $\pi = (1 2 \dots n) \in S_n$ we see that $|T_{\pi}\rangle = |T_n\rangle$ as defined in Eq. (3.2.1).

We also note that we recover the notion of uniform PEPS on a grid by choosing appropriate permutations. For instance, for $m = 2$, and a grid of size $n_1 \times n_2$ this would correspond to using the permutations

$$\begin{aligned} \pi_1 &= (1 2 \dots n_1)(n_1 + 1 n_1 + 2 \dots 2n_1) \dots ((n_2 - 1)n_1 + 1 (n_2 - 1)n_1 + 2 \dots n_2 n_1) \\ \pi_2 &= (1 n_1 + 1 \dots (n_2 - 1)n_1 + 1)(2 n_1 + 2 \dots (n_2 - 1)n_1 + 2)(n_1 2n_1 \dots n_2 n_1). \end{aligned}$$

This yields (upon appropriately identifying the copies of \mathbb{H}_{phys}) an equivalence $|T_{n_1, n_2}\rangle = |T_{(\pi_1, \pi_2)}\rangle$.

As in the MPS case, we have a ‘gauge group’ acting on the tensor. We can now act with a different group element along each direction $k \in [m]$.

Definition 3.3.3 (Gauge action). We define the *gauge action* of $\mathbf{g} \in G = GL(D_1) \times \cdots \times GL(D_m)$, where $\mathbf{g} = (g_1, \dots, g_m)$, on $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ as

$$\mathbf{g} \cdot T = \left((g_1 \otimes \dots \otimes g_m) T^{(i)} (g_1^{-1} \otimes \dots \otimes g_m^{-1}) \right)_{i=1}^d.$$

If we think of T as a quantum state $|T\rangle$ in $(\bigotimes_{k=1}^m \mathbb{H}_{k,1} \otimes \mathbb{H}_{k,2}) \otimes \mathbb{H}_{\text{phys}}$, the gauge action can be written as

$$\mathbf{g} \cdot |T\rangle = \left(\left(\bigotimes_{k=1}^m g_k \otimes g_k^{-T} \right) \otimes I \right) |T\rangle.$$

As in the MPS case, it is easy to see that this action keeps the associated PEPS invariant. By continuity, this is also true after taking limits, giving rise to the following lemma.

Lemma 3.3.4. For every $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$, $G = GL(D_1) \times \cdots \times GL(D_m)$, if $T' \in \overline{G \cdot T}$, then for all $\pi \in S_n^m$

$$|T_\pi\rangle = |T'_\pi\rangle.$$

and in particular

$$P_{\pi,i}(T) = P_{\pi,i}(T').$$

In other words, the coefficient functions $P_{\pi,i}$ are polynomials in the invariant ring $\mathbb{C}[\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d]^G$. We have a corresponding notion of gauge equivalence.

Definition 3.3.5 (Gauge equivalence). Let $S, T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ be two matrix tuples. Let $G = GL(D_1) \times \cdots \times GL(D_m)$. We say that S and T are *gauge equivalent* if and only if $\overline{G \cdot S} \cap \overline{G \cdot T} \neq \emptyset$.

3.3.2. Minimal canonical form

We consider uniform PEPS in m spatial dimensions with bond dimensions D_1, \dots, D_m and physical dimension d . We denote the gauge group by $G = GL(D_1) \times \cdots \times GL(D_m)$. We denote by $K = U(D_1) \times \cdots \times U(D_m) \subset G$ the unitary subgroup. We can now follow exactly the same approach as in the MPS case to define the minimal canonical form, and the same general results from geometric invariant theory allow us to prove its basic properties.

Definition 3.3.6 (Minimal canonical form PEPS). We say $T_{\min} \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is a minimal canonical form of $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ if it is an element of minimal norm in the orbit closure $\overline{G \cdot T}$, i.e.,

$$T_{\min} = \operatorname{argmin} \{ \|S\| : S \in \overline{G \cdot T} \}.$$

We say $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is *in canonical form* if it is a minimal canonical form for itself, i.e. an element of minimal norm in $\overline{G \cdot T}$.

3. The minimal canonical form of a tensor network

The norm considered in the definition is, as in the MPS case, the Euclidean norm of T (or $|T\rangle$):

$$\|T\| = \sqrt{\langle T|T\rangle} = \left(\sum_{i=1}^d \text{Tr} \left[(T^{(i)})^\dagger T^{(i)} \right] \right)^{1/2}.$$

The minimal canonical form is not uniquely defined, but it is unique up to the action by the unitary group $K = U(D_1) \times \cdots \times U(D_m)$:

Theorem 3.3.7 (Minimal canonical form). *Let $S, T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$. Then the following are equivalent:*

- (i) S and T have a common minimal canonical form.
- (ii) If S_{\min} and T_{\min} are minimal canonical forms for S and T , then $K \cdot S_{\min} = K \cdot T_{\min}$.
- (iii) S and T are gauge equivalent, i.e., $\overline{G \cdot S} \cap \overline{G \cdot T} \neq \emptyset$.

Proof. This is an immediate consequence of Theorems 2.3.7 and 2.4.4. \square

Recall that if $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is a PEPS tensor, we saw in Eq. (3.3.1) that we may consider it as a quantum state $|T\rangle$. For each ‘direction’ $k \in [m]$ we have two virtual Hilbert spaces $\mathbb{H}_{k,1}$ and $\mathbb{H}_{k,2}$ of dimension D_k and there is the physical Hilbert space \mathbb{H}_{phys} of dimension d . We denote by $\rho_{k,j}$ the reduced state of $\rho = |T\rangle\langle T|$ on $\mathbb{H}_{k,j}$.

The characterization of minimum norm vectors as critical norm vectors in Theorem 2.4.4 can be used to give a condition for a tensor to be in minimal canonical form. To find this condition we perform a computation similar to the MPS case. We identify $i\text{Lie}(K)$ with $\text{Herm}(D_1) \times \cdots \times \text{Herm}(D_m)$ and compute for $\mathbf{X} = (X_1, \dots, X_m) \in \text{Herm}(D_1) \times \cdots \times \text{Herm}(D_m)$

$$\begin{aligned} & \partial_{t=0} \| (e^{tX_1}, \dots, e^{tX_m}) \cdot T \|^2 \\ &= \partial_{t=0} \text{Tr} \left[\sum_{i=1}^d (e^{2tX_1} \otimes \dots \otimes e^{2tX_m}) T^{(i)} (e^{-2tX_1} \otimes \dots \otimes e^{-2tX_m}) (T^{(i)})^\dagger \right] \\ &= 2 \sum_{k=1}^m \text{Tr} \left[I_{D_1} \otimes \dots \otimes X_k \otimes \dots \otimes I_{D_m} \left(\sum_{i=1}^d T^{(i)} (T^{(i)})^\dagger - (T^{(i)})^\dagger T^{(i)} \right) \right] \quad (3.3.4) \\ &= 2 \sum_{k=1}^m \text{Tr} \left[X_k \left(\rho_{k,1} - \rho_{k,2}^T \right) \right]. \end{aligned}$$

Theorem 3.3.8 (Characterization). *Let $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$. Then T is in minimal canonical form if and only if $\|\mathbf{g} \cdot T\| \geq \|T\|$ for all $\mathbf{g} \in G$. This is the case if and only if the reduced density matrices of $\rho = |T\rangle\langle T|$ on the virtual bonds are the same in each direction, up to a transpose:*

$$\rho_{k,1} = \rho_{k,2}^T \quad (\forall k \in [m]) \quad (3.3.5)$$

Proof. By Theorem 2.4.4, T is in minimal canonical form if and only if it is critical, which means that the derivative in Eq. (3.3.4) should vanish for all \mathbf{X} . This is equivalent to $\rho_{k,1} = \rho_{k,2}^T$ for all $k \in [m]$. \square

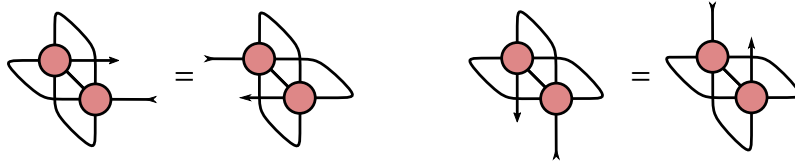


Figure 3.8.: *Minimal canonical form PEPS*: Graphical version of the conditions $\rho_{1,1} = \rho_{1,2}^\top$ and $\rho_{2,1} = \rho_{2,2}^\top$ from Theorem 3.3.8.

These conditions are illustrated in Fig. 3.8 for $m = 2$. Without the framework of invariant theory, it is not clear that one can indeed transform any tensor by gauge transformations to satisfy the conditions in Theorem 3.3.8. This is an important difference with earlier proposals for canonical forms for PEPS. For instance, [PMV15] proposes a canonical form based on a similar (but different) condition. However, in that case, it is not clear that such a canonical form indeed exists for any tensor.

Both Theorem 3.3.7 and Theorem 3.3.8, giving the “uniqueness” of the canonical form and its algebraic characterization respectively, only require situations in which one is already interested in analyzing tensors related by gauge transformations. Reducing to such a situation is the goal of the *Fundamental Theorems*. For MPS we already saw such fundamental theorems, in particular Theorem 3.2.14, which apply to general MPS.

For PEPS the situation is more complicated, but for important special cases, fundamental theorems are known. In particular, fundamental theorems are known for the family of *normal* tensors [CPSV17], proven for the uniform 2D case in [PSG+10], and extended to the general case in [MGP+18].

To define normal tensors, we first recall the notion of an *injective* PEPS tensor. A tensor $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is injective if it is injective as a map from the virtual legs to the physical legs, i.e. if it is injective as a $d \times D_1^2 \dots D_m^2$ matrix. The tensor T is *normal* if it is injective after blocking together a number of copies to a single new tensor. Let us explain what we mean by ‘blocking’. Given $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ we can contract $n = n_1 \dots n_m$ copies of T on a rectangular lattice of size $n_1 \times \dots \times n_m$ sites to obtain a new tensor \tilde{T} with physical dimension d^n and bond dimensions $D_1^{n_2 \dots n_m}, D_2^{n_1 n_3 \dots n_m}, \dots, D_m^{n_1 \dots n_{m-1}}$. The tensor T is normal if there exists some blocking such that the resulting tensor \tilde{T} is injective.

Hence in the normal case, which is a generic condition, Theorem 3.3.7 and Theorem 3.3.8 together with the Fundamental Theorem of [PSG+10] already apply to show the following statement (for simplicity we only write down the two-dimensional case):

Corollary 3.3.9. *Two normal tensors T and S in $\text{Mat}_{D_1 D_2 \times D_1 D_2}^d$ define the same state in all $n_1 \times n_2$ grids, i.e. $|T_{n_1, n_2}\rangle = |S_{n_1, n_2}\rangle$ for all $n_1, n_2 \in \mathbb{N}$, if and only if their corresponding minimal canonical forms S_{\min} and T_{\min} are related by local unitary gauges: $S_{\min} = \mathbf{U} \cdot T_{\min}$ for a suitable unitary $\mathbf{U} \in U(D_1) \times U(D_2)$.*

Moreover, we will see below in Proposition 3.3.20 that the orbit of a normal tensor is always closed. However, this is not the end of the story. There are other (non-normal) tensors which define the same state in all $n_1 \times n_2$ grids, but are nevertheless *not* related by a gauge transformation. An explicit example appears in [MGSC18], in the context of 2D SPT phases. We provide the example here:

Example 3.3.10. The idea of the example is simple but ingenious. Take pairs of MPS normal tensors A and B so that $|A_4\rangle = |B_4\rangle$ but $|A_j\rangle \neq |B_j\rangle$ for all $j > 4$.⁴ The explicit examples of [MGSC18] have physical dimension 2 and are given by the matrices:

$$A^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \quad A^{(2)} = \begin{bmatrix} 24 & -10 \\ 17 & -3 \end{bmatrix},$$

where $B^{(1)} = A^{(1)}$ and $B^{(2)} = -A^{(2)}$.

Now, in each vertex of a two dimensional grid, place four qubits and, by joining each one of those qubits with the closest one in each of the nearest neighbor sites, fill in the lattice with a set of non-overlapping plaquettes \mathcal{P} . The states we are interested are $|M_A\rangle = \bigotimes_{p \in \mathcal{P}} |A_4\rangle_p$ and $|M_B\rangle = \bigotimes_{p \in \mathcal{P}} |B_4\rangle_p$. It is now obvious how to define the associated PEPS tensors M_A and M_B for the vertices. Just take, with the appropriate identification of indices, $M_A = A^{\otimes 4}$, $M_B = B^{\otimes 4}$ (recall that each vertex contains four qubits and therefore the physical dimension is 16). It is shown in [MGSC18] that tensors M_A, M_B are not in the same $GL_4 \times GL_4$ orbit. One can indeed show that the closure of their orbits do not intersect. One possibility is just to realize that, because of the symmetry of the tensors M_A and M_B , they are already in minimal canonical form, and therefore their orbits are already closed. The other possibility is to compare M_A and M_B in different contraction graphs Γ . It is easy to find some Γ for which the length of some of the plaquettes are larger than 4 and then the fact that $|A_j\rangle \neq |B_j\rangle$ for $j > 4$ implies that the associated states $|M_{A,\Gamma}\rangle$ and $|M_{B,\Gamma}\rangle$ are different, which in turn implies that the orbits of M_A and M_B cannot intersect.

3.3.3. Fundamental theorem and invariant theory of uniform PEPS

This example makes clear that we have to change perspective to derive a Fundamental Theorem which is an analog to the MPS one (Theorem 3.2.14). Instead of starting with the condition $|S_{n_1, n_2}\rangle = |T_{n_1, n_2}\rangle$ for all n_1, n_2 , and asking how the tensors S and T are related, we start with the condition that S and T are gauge equivalent, and we ask how we can characterize this based on the corresponding tensor network states. It turns out that we need to compare the states not just on grids, but on arbitrary contraction graphs. That is, the appropriate conditions is $|S_\pi\rangle = |T_\pi\rangle$ for tuples of permutations π .

Additionally, for MPS we found that it suffices to consider systems of size at most D^2 (Theorem 3.2.14) or even $\tilde{O}(D)$ (Remark 3.2.15). For $m \geq 2$ we prove a similar bound, but now we need a system size exponential in D (and we show below, in Proposition 3.3.15, that this exponential dependence cannot be avoided). Formally, we have the following weak version of a Fundamental Theorem, illustrated in Fig. 3.3.

Theorem 3.3.11 (Fundamental Theorem for PEPS). *Let $S, T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$. Then the following are equivalent:*

- (i) *The G -orbit closures of S and T intersect, i.e., $\overline{G \cdot S} \cap \overline{G \cdot T} \neq \emptyset$.*

⁴It is only proven in [MGSC18] that $|A_5\rangle \neq |B_5\rangle$, but since it is also shown that both A and B become injective when blocking two sites, known bounds for the fundamental theorem [MGP+18] imply already that if $|A_j\rangle = |B_j\rangle$ for any $j \geq 6$, then A and B would be gauge-related and then $|A_5\rangle = |B_5\rangle$.

- (ii) $|S_{\pi_1, \dots, \pi_m}\rangle = |T_{\pi_1, \dots, \pi_m}\rangle$ for all $\pi_k \in S_r$ for all $r \in \mathbb{N}$.
- (iii) $|S_{\pi_1, \dots, \pi_m}\rangle = |T_{\pi_1, \dots, \pi_m}\rangle$ for all $\pi_k \in S_r$ for $r \leq \exp(cmD^2 \log D)$ where $D = \max\{D_1, \dots, D_m\}$ and c is a constant.

To prove this result, we start with the following lemma (which is a basic result in invariant theory [KP96, §4.6]), which allows us to reduce the study of invariant polynomials $\mathbb{C}[\text{Mat}_{D \times D}^d]^G$ to the study of multilinear invariant polynomials. While the result is a basic one, it is a key component in proving a number of *first fundamental theorems* in invariant theory, see [KP96] for more details.

Lemma 3.3.12. *For any subgroup $G \subset GL(D)$, any polynomial P in the ring of invariant polynomials $\mathbb{C}[\text{Mat}_{D \times D}^d]^G$ can be written as a linear combination of multihomogeneous invariant polynomials $P_{\mathbf{n}}$ of some multidegree $\mathbf{n} = (n_1, \dots, n_d)$, each of which can be written as*

$$P_{\mathbf{n}}(M^{(1)}, \dots, M^{(d)}) = Q(\underbrace{M^{(1)}, \dots, M^{(1)}}_{n_1 \text{ times}}, \dots, \underbrace{M^{(d)}, \dots, M^{(d)}}_{n_d \text{ times}}), \quad (3.3.6)$$

where Q is a multilinear G -invariant polynomial in $\mathbf{n} = \sum_{i=1}^d n_i$ matrix variables.

Proof. Let $P = P(M^{(1)}, \dots, M^{(d)}) \in \mathbb{C}[\text{Mat}_{D \times D}^d]^G$. First we show that we may assume that P is multihomogeneous, i.e., homogeneous of some degree n_i in each matrix variable $M^{(i)}$. Indeed, we can write

$$P(M^{(1)}, \dots, M^{(d)}) = \sum_{\mathbf{n}=(n_1, \dots, n_d)} P_{\mathbf{n}}(M^{(1)}, \dots, M^{(d)}),$$

where $P_{\mathbf{n}}$ is homogeneous of degree n_i in the matrix variable $M^{(i)}$. Since the space of homogeneous polynomials of multidegree \mathbf{n} is invariant under $GL(D)$, and spaces of different multidegree are linearly independent, each $P_{\mathbf{n}}$ is G -invariant. Thus we may without loss of generality assume that $P = P_{\mathbf{n}}$. Next, we reduce to *multilinear* invariants of some possibly larger number of matrices, as follows. Consider $P(M^{(1,1)} + \dots + M^{(1,n_1)}, \dots, M^{(d,1)} + \dots + M^{(d,n_d)})$, a polynomial in formal matrix variables $M^{(i,j)}$ for $i \in [d]$ and $j \in [n_i]$, and write

$$\begin{aligned} & P(M^{(1,1)} + \dots + M^{(1,n_1)}, \dots, M^{(d,1)} + \dots + M^{(d,n_d)}) \\ &= \sum_{\mathbf{h}=(h_{1,1}, \dots, h_{d,n_d})} P_{\mathbf{h}}(M^{(1,1)}, \dots, M^{(d,n_d)}), \end{aligned}$$

where $P_{\mathbf{h}}$ is homogeneous of degree $h_{i,j}$ in each matrix variable $M^{(i,j)}$. Now note that for all $t_{1,1}, \dots, t_{d,n_d}$,

$$\begin{aligned} & P(t_{1,1}M^{(1,1)} + \dots + t_{1,n_1}M^{(1,n_1)}, \dots, t_{d,1}M^{(d,1)} + \dots + t_{d,n_d}M^{(d,n_d)}) \\ &= \sum_{\mathbf{h}=(h_{1,1}, \dots, h_{d,n_d})} \mathbf{t}^{\mathbf{h}} P_{\mathbf{h}}(M^{(1,1)}, \dots, M^{(d,n_d)}), \end{aligned} \quad (3.3.7)$$

so if we take $M^{(i,j)} \equiv M^{(i)}$ for all $i \in [d]$ and $j \in [n_i]$ we have

$$P(t_{1,1}M^{(1)} + \dots + t_{1,n_1}M^{(1)}, \dots, t_{d,1}M^{(d)} + \dots + t_{d,n_d}M^{(d)})$$

3. The minimal canonical form of a tensor network

$$= \sum_{\mathbf{h}=(h_{1,1},\dots,h_{d,n_d})} \mathbf{t}^{\mathbf{h}} P_{\mathbf{h}}(\underbrace{M^{(1)},\dots,M^{(1)}}_{n_1 \text{ times}}, \dots, \underbrace{M^{(d)},\dots,M^{(d)}}_{n_d \text{ times}}).$$

On the other hand, by multihomogeneity,

$$\begin{aligned} & P(t_{1,1}M^{(1)} + \dots + t_{1,n_1}M^{(1)}, \dots, t_{d,1}M^{(d)} + \dots + t_{d,n_d}M^{(d)}) \\ &= (t_{1,1} + \dots + t_{1,n_1})^{n_1} \dots (t_{d,1} + \dots + t_{d,n_d})^{n_d} P(M^{(1)}, \dots, M^{(d)}) \\ &= \sum_{\mathbf{h}=(h_{1,1},\dots,h_{d,n_d})} \binom{n_1}{h_{1,1} \dots h_{1,n_1}} \dots \binom{n_d}{h_{d,1} \dots h_{d,n_d}} \mathbf{t}^{\mathbf{h}} P(M^{(1)}, \dots, M^{(d)}). \end{aligned}$$

Comparing coefficients and specializing to $\mathbf{h} = (1, \dots, 1)$, we find that

$$P(M^{(1)}, \dots, M^{(d)}) = \frac{1}{n_1! \dots n_d!} P_{1,\dots,1}(\underbrace{M^{(1)}, \dots, M^{(1)}}_{n_1 \text{ times}}, \dots, \underbrace{M^{(d)}, \dots, M^{(d)}}_{n_d \text{ times}}).$$

Note that $P_{1,\dots,1}$ is a multilinear polynomial in $\sum_{i=1}^d n_i$ matrix variables. Since the left-hand side of Eq. (3.3.7) is G -invariant, we may also assume that $P_{1,\dots,1}$ is G -invariant. \square

We now return to our setting, where $G = GL(D_1) \times \dots \times GL(D_m)$, and use this lemma to prove.

Proposition 3.3.13. *The ring of invariant polynomials $\mathbb{C}[\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d]^G$ is generated by functions $P_{\pi,i}$ as in Eq. (3.3.3) for $n \leq \exp(cmD^2 \log(mD))$ where $D = \max\{D_1, \dots, D_m\}$ and $c > 0$ is a universal constant.*

Proof. Let $P = P(T^{(1)}, \dots, T^{(d)}) \in \mathbb{C}[\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d]^G$. By Lemma 3.3.12 with $D = D_1 \dots D_m$ we may reduce to the case where $P = P_{\mathbf{n}}$ for some $\mathbf{n} = (n_1, \dots, n_d)$, and we can write

$$P(T^{(1)}, \dots, T^{(d)}) = \langle \mathbf{R}, \underbrace{T^{(1)} \otimes \dots \otimes T^{(1)}}_{n_1 \text{ times}} \otimes \underbrace{T^{(2)} \otimes \dots \otimes T^{(2)}}_{n_2 \text{ times}} \otimes \dots \otimes \underbrace{T^{(d)} \otimes \dots \otimes T^{(d)}}_{n_d \text{ times}} \rangle,$$

where $\langle \cdot, \cdot \rangle$ is the trace inner product and where

$$\mathbf{R} \in \left(\text{End}(\mathbb{C}^{D_1} \otimes \dots \otimes \mathbb{C}^{D_m})^{\otimes n} \right)^G.$$

The total degree is given by $n = \sum_{i=1}^d n_i$. Now note that

$$\begin{aligned} & \left(\text{End}(\mathbb{C}^{D_1} \otimes \dots \otimes \mathbb{C}^{D_m})^{\otimes n} \right)^G \\ & \cong \text{End}((\mathbb{C}^{D_1})^{\otimes n})^{GL(D_1)} \otimes \dots \otimes \text{End}((\mathbb{C}^{D_m})^{\otimes n})^{GL(D_m)} \\ & \cong \mathbb{C}[\mathbf{R}_{\pi_1} : \pi_1 \in S_n] \otimes \dots \otimes \mathbb{C}[\mathbf{R}_{\pi_m} : \pi_m \in S_n] \\ & \cong \mathbb{C}[\mathbf{R}_{\pi_1} \otimes \dots \otimes \mathbf{R}_{\pi_m} : \pi_1, \dots, \pi_m \in S_n] \end{aligned}$$

where we denote by \mathbf{R}_{π_k} the operator acting on $(\mathbb{C}^{D_k})^{\otimes n}$ permuting the n copies of \mathbb{C}^{D_k} according to π_k . Thus, \mathbf{R} is a linear combination of elements of the

form $R_\pi = R_{\pi_1} \otimes \dots \otimes R_{\pi_m}$, for $\pi = (\pi_1, \dots, \pi_m)$. We conclude that the ring of invariant polynomials $\mathbb{C}[\text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d]^G$ is generated as a vector space by the polynomial functions $P_{\pi, i}$ as in Eq. (3.3.3) for $\pi \in S_n^m$ and $n \in \mathbb{N}$. In particular, the invariant polynomials of degree at most r are spanned by the $P_{\pi, i}$ for $\pi \in S_n^m$ and $i \in [d]^n$ for $n \leq r$.

We now use general results in invariant theory to bound the degree necessary to generate the invariant ring as an algebra. For convenience, we write $V := \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}$, so we are interested in degree bounds for the action of $G = \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$ on $V^d = \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$. We first appeal to a classical theorem by Weyl [Wey46, II.5 Thm. 2.5.A] which states that if $d > \dim(V)$, a generating set of invariants for V^d can be obtained by acting with $\text{GL}(d)$ on a generating set for $\mathbb{C}[V^{\dim(V)}]^G \hookrightarrow \mathbb{C}[V^d]^G$ (cf. [KP96, §7.1]). In particular, any degree bound for $d = \dim(V)$ also applies to $d > \dim(V)$. Accordingly, we may assume without loss of generality that $d \leq \dim(V)$. Next, we observe that since we act by simultaneous conjugation, the invariants for the action of G are the same as for $G' := \text{SL}(D_1) \times \dots \times \text{SL}(D_m)$, so we can restrict to the latter. By results of Derksen [Der00] the ring of invariants is generated by invariant polynomials of degree at most

$$r \leq \frac{3}{8} \dim(V^d) (H^{t - \dim(G')} A^{\dim(G')})^2 \quad (3.3.8)$$

where t, H, A are integers computed as follows. We think of G' as being embedded in $\bigoplus_{k=1}^m \text{Mat}_{D_k \times D_k} \cong \mathbb{C}^t$, with $t = \sum_{k=1}^m D_k^2$. Then G' is defined as the common zero set of the polynomials $\det(g_k) - 1$ for $k \in [m]$. The integer H is the maximal degree of these polynomials, i.e., $H = \max_k D_k$. If one fixes an arbitrary basis of V^d , the matrix entries of the representation of G' are polynomial functions of the coordinates of \mathbb{C}^t (that is, the entries of the g_k). The integer A is the maximal degree of these polynomials. To compute it, note that $(g_1, \dots, g_m) \in G'$ acts on a matrix tuple $T = (T^{(i)})_{i=1}^d \in V^d$ by simultaneous conjugation by $g_1 \otimes \dots \otimes g_m$. Thus, we left multiply each matrix $T^{(i)}$ with $g_1 \otimes \dots \otimes g_m$, the entries of which are polynomials of degree m in the entries of the g_k , and we right multiply each $T^{(i)}$ with

$$g_1^{-1} \otimes \dots \otimes g_m^{-1} = \text{adj}(g_1) \otimes \dots \otimes \text{adj}(g_m), \quad (3.3.9)$$

where $\text{adj}(g_k)$ is the adjugate matrix of g_k (here we used that $g_k \in \text{SL}(D_k)$, so that we did not have to divide by the determinant when computing the inverse); since the entries of the adjugate matrix are given by cofactors of g_k and hence have degree $D_k - 1$, the entries of (3.3.9) are polynomials of degree $\sum_{k=1}^m (D_k - 1)$. Therefore, each matrix entry of the representation of G' is a polynomial of degree $A = m + \sum_{k=1}^m (D_k - 1) = \sum_{k=1}^m D_k$.

Evaluating Eq. (3.3.8) with $\dim(V^d) = d \prod_{k=1}^m D_k^2$, $d \leq \dim(V)$, $\dim(G') = \sum_{k=1}^m (D_k^2 - 1)$, $H = \max_k D_k$, $t = \sum_{k=1}^m D_k^2$ and $A = \sum_{k=1}^m D_k$ shows that we can bound the required degree by

$$n \leq \frac{3}{8} \left(d \prod_{k=1}^m D_k^2 \right) \left(\left(\max_k D_k \right)^m \left(\sum_{k=1}^m D_k \right)^{\sum_{k=1}^m (D_k^2 - 1)} \right)^2 \leq \exp(cmD^2 \log(mD))$$

for some universal constant $c \geq 0$. □

3. The minimal canonical form of a tensor network

Proof of Theorem 3.3.11. It is clear that (i) \Rightarrow (ii) \Rightarrow (iii). The fact that (iii) \Rightarrow (i) is a consequence of Proposition 3.3.13 and Theorem 2.3.7. \square

Corollary 3.3.14 (Lifting symmetries). *Suppose that $S, T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ are in minimal canonical form and $u \in U(d)$ is a unitary such that $u^{\otimes n} |S_\pi\rangle = |T_\pi\rangle$ for all $\pi \in S_n^m$ and $n \in \mathbb{N}$. Then there exist unitaries $U_k \in U(D_k)$ such that $(I \otimes u) |S\rangle = ((\bigotimes_{k=1}^m U_k \otimes \bar{U}_k) \otimes I) |T\rangle$.*

Proof. Let $S' \in \text{Mat}_{D \times D}^d$ be the matrix tuple defined by

$$|S'\rangle := (I \otimes u) |S\rangle.$$

Then S' is also in minimal canonical form, since u is unitary and hence we have $\|g \cdot S\| = \|g \cdot S'\|$ for all $g \in G$. Moreover, by construction it holds that

$$|S'_\pi\rangle = u^{\otimes n} |S_\pi\rangle = |T_\pi\rangle$$

for all $\pi \in S_n^m$ and $n \in \mathbb{N}$. Thus Theorem 3.3.11 shows that S' and T are gauge equivalent, and it follows from Theorem 3.3.7 that there exist unitary gauge transformations $U_k \in U(D_k)$ such that $(I \otimes u) |S\rangle = ((\bigotimes_{k=1}^m U_k \otimes \bar{U}_k) \otimes I) |T\rangle$. \square

The degree bounds in Proposition 3.3.13 are a direct consequence of deep and completely general results in invariant theory. These bounds are in general not necessarily sharp. As an example, the degree bounds obtained in this way for the MPS case are still exponential, while we know from Theorem 3.2.13 that in this special case we have a degree bound of D^2 . Moreover, we know from Remark 3.2.15 that in this case invariants of degree $\tilde{O}(D)$ already suffice to determine whether two MPS tensors are gauge equivalent.

However, this is quite special for one spatial dimension. For PEPS with spatial dimension $m \geq 2$, we now show that one in general needs to consider invariants of degree exponential in the bond dimension in order to decide whether two PEPS tensors are gauge equivalent (even if one is the zero tensor). For convenience we take $m = 2$, $D_1 = D_2 = D$, and $d = 1$ (that is, the tensor networks defined by the PEPS tensors are scalars).

Proposition 3.3.15 (Degree lower bound). *There exists a function $n_{\min}(D) = e^{\Omega(D)}$ and, for every D , a tensor $T \in \text{Mat}_{D^2 \times D^2}$ with the following properties:*

- (i) *For any invariant polynomial $P \in \mathbb{C}[\text{Mat}_{D^2 \times D^2}]^{\text{GL}(D) \times \text{GL}(D)}$ of degree less than $n_{\min}(D)$, we have $P(T) = P(0)$.*
- (ii) *There exists an invariant polynomial P of degree $n_{\min}(D)$ such that $P(T) \neq P(0)$. In particular, we have $0 \notin \overline{G \cdot T}$, meaning that T is not gauge equivalent to the zero tensor.*

In particular, the ring of invariant polynomials $\mathbb{C}[\text{Mat}_{D^2 \times D^2}^d]^{\text{GL}(D) \times \text{GL}(D)}$ for any $d \geq 1$ is not generated by the polynomials of degree $n < n_{\min}(D)$.

Proof. The last statement of the proposition is an immediate consequence of the described properties of T . Indeed, if the ring of invariants were generated by invariant polynomials of degree smaller than $n_{\min}(D)$, then $P(T) = P(0)$ for all such

polynomials P would imply that $P(T) = P(0)$ for all invariant polynomials P – but we know that $P(T) \neq P(0)$ for at least one invariant polynomial of degree $n_{\min}(D)$.

We will explicitly construct a tensor $T \in \text{Mat}_{D^2 \times D^2}$. For $d = 1$ and for $\pi \in S_n^m$ and $\mathbf{1} = (1, \dots, 1)$ we abbreviate $P_{\pi, \mathbf{1}} = P_\pi$. Since the P_π for $\pi \in S_n^{\times 2}$ for $n < r$ are homogeneous and span the degree r polynomials in $\mathbb{C}[\text{Mat}_{D^2 \times D^2}]^{\text{GL}(D) \times \text{GL}(D)}$ it suffices to show that $P_\pi(T) = 0$ for $\pi \in S_n^{\times 2}$ for $n < n_{\min}$, while there exists some $\pi \in S_n^{\times 2}$ for $n = n_{\min}$ such that $P_\pi(T) \neq 0$. We will take $n_{\min} = 2^D + 2^{D-1} - 2$.

To explain the construction and the argument we start with a construction where we allow the physical dimension d to grow with D , and we construct a tensor $S \in \text{Mat}_{D^2 \times D^2}^{2^{D-1}}$ with certain properties. Then, we will use a trick to reduce the physical dimension. Let $\{|j\rangle\}_{j=0}^{D-1}$ denote the standard basis of \mathbb{C}^D . We choose the tensor S as follows:

$$S^{(1)} = |0\rangle \langle 1| \otimes |0\rangle \langle 1|, \quad S^{(2j)} = |j\rangle \langle 0| \otimes |0\rangle \langle j|, \quad S^{(2j+1)} = |0\rangle \langle j+1| \otimes |j\rangle \langle j+1|$$

for $j = 1, \dots, D-1$ and where the index j should be read modulo D (so $|D\rangle = |0\rangle$). We will now argue that on the one hand, for all $\mathbf{i} = (i_1, \dots, i_n) \in [2D-1]^n$ and $n < 2^D + 2^{D-1} - 2$ we have $P_{\pi, \mathbf{i}}(S) = 0$ for all π , while on the other hand for $n = 2^D + 2^{D-1} - 2$ there is some π and $\mathbf{i} = (i_1, \dots, i_n)$ with $P_{\pi, \mathbf{i}}(S) \neq 0$.

We start by showing that if $\mathbf{i} = (i_1, \dots, i_n)$ with $n < 2^D + 2^{D-1} - 2$ then we have $P_{\pi, \mathbf{i}}(S) = 0$. To conveniently reason about contractions in the tensor network picture we will name the four virtual legs of the tensors as follows:

$$|\text{left}\rangle \langle \text{right}| \otimes |\text{down}\rangle \langle \text{up}|$$

and call the two directions ‘horizontal’ and ‘vertical’. In the tensor network picture, we observe that for each even $i = 2j$ one can only contract the upper leg of $S^{(2j)}$ along the vertical direction with a copy of $S^{(2j+1)}$ in order for the result to be nonzero. That is, if we have $i_k = i$ even, then π_2 must map k to l where $i_l = i + 1$. Similarly, for $i = 2j + 1 < 2D - 1$ odd we need to contract the right leg of $S^{(2j+1)}$ with the left leg of a copy of $S^{(2j+2)}$ in the horizontal direction and its upper leg with a copy of $S^{(2j+3)}$ in the vertical direction. Together these conditions imply that if n_i denotes the number of copies of $S^{(i)}$ one requires in order for the contraction to be nonzero, we have $n_{i+2} \geq n_{i+1} + n_i$ for $i < 2D - 1$ odd and $n_{i+1} \geq n_i$ for $i \leq 2D$ even. By similar reasoning, for even $i = 2j$, the left leg of a copy of $S^{(2j)}$ needs to be contracted in the horizontal direction with a copy of $S^{(2j-1)}$, and for odd $i = 2j + 1 > 1$, the down leg of a copy of $S^{(2j+1)}$ needs to be contracted in the vertical direction with a copy of $S^{(2j)}$ or $S^{(2j-1)}$. This implies that if $n_i \neq 0$ for $i \geq 2$ we also need either n_{i-1} or n_{i-2} to be nonzero and in particular $n_1 \geq 1$.

Solving the recursion with $n_1 \geq 1$ gives $n_{2i+1} \geq 2^i$ and $n_{2i} \geq 2^{i-1}$ for $i = 1, \dots, D-1$. We then have

$$n = \sum_{i=1}^{2D-1} n_i \geq 2^D + 2^{D-1} - 2.$$

On the other hand, it is easy to see that if we take n_i copies of $S^{(i)}$ with $n_1 = 1$, $n_{2i} = 2^{i-1}$ and $n_{2i+1} = 2^i$ we can indeed contract to something nonzero.

3. The minimal canonical form of a tensor network

Now, to prove the proposition, we adapt the previous construction to $d = 1$. We construct $T \in \text{Mat}_{D^2 \times D^2}$ as

$$T = T^{(1)} = \sum_{i=1}^{2D-1} S^{(i)}.$$

Consider some arbitrary $\pi \in S_n^m$. We may expand $T = \sum_i S^{(i)}$ for each copy of T to find

$$P_\pi(T) = \sum_{i \in [d]^n} P_{\pi, i}(S).$$

By construction of S , each $P_{\pi, i}(S)$ is either zero or one, proving that $P_\pi(T) \neq 0$ if and only if there is some $i = (i_1, \dots, i_n)$ such that $P_{\pi, i}(S) \neq 0$.

By our previous arguments for S this implies that for all $n < 2^D + 2^{D-1} - 2$ and $\pi \in S_n^m$ we have $P_\pi(T) = 0$, but that for $n = 2^D + 2^{D-1} - 2$ we can find some $\pi \in S_n^m$ such that $P_\pi(T) \neq 0$. \square

Remark 3.3.16. *The argument of Proposition 3.3.15 can be extended to $m > 2$. We define a generalization of $S \in \text{Mat}_{D^m \times D^m}^{m(D-1)+1}$ as follows: for $i = 1, \dots, D-1$ and $j = 1, \dots, m-1$ set*

$$S^{(1)} = (|0\rangle \langle 1|)^{\otimes m}, \quad S^{(m(i-1)+j+1)} = (|0\rangle \langle 0|)^{\otimes (j-1)} \otimes |i\rangle \langle 0| \otimes (|0\rangle \langle i|)^{\otimes (m-j)}.$$

and

$$S^{(mi+1)} = (|0\rangle \langle i+1|)^{\otimes (m-1)} \otimes |i\rangle \langle i+1|.$$

Note that as before we interpret the basis states modulo D , i.e., $|D\rangle = |0\rangle$. Then again define $T \in \text{Mat}_{D^m \times D^m}$ by

$$T = T^{(1)} = \sum_{i=1}^{m(D-1)+1} S^{(i)}$$

Essentially the same argument yields

$$n_{\min} = 1 + \sum_{i=1}^{D-1} 2^{i(m-1)} \sum_{j=0}^{m-1} 2^j = \exp(\Omega(mD))$$

so the degree lower bound also scales exponentially in m .

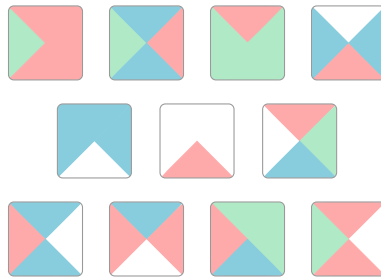
We note here that proving degree lower bounds is not often an easy task, and in literature often has to employ rather involved and indirect techniques to get exponential lower bounds even in very familiar cases, see e.g., [DM20b; DM22]. The technique we use above is far more straightforward and explicit even though the setting we study here is somewhat similar to some of the cases handled in the aforementioned papers.

3.3.4. Two-dimensional tensor networks, tilings and topology

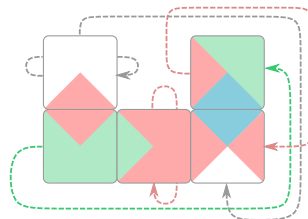
Consider the following question: given a PEPS tensor T in two spatial dimensions, determine whether there exist n_1, n_2 such that the associated state $|T_{n_1, n_2}\rangle$ on a rectangular periodic lattice of size $n_1 \times n_2$ is nonzero. This problem is undecidable, see [SMG+20]. The proof of the undecidability given in [SMG+20] is by reducing to the problem of the existence of a periodic tiling given some set of tiles. Given a set of square tiles where each edge of the tile is associated to one of D boundary colors, the question is whether there exists a tiling (meaning that the boundary colors of adjacent tiles match) which is periodic. Equivalently, this gives a tiling of the two-dimensional torus. It is known that the existence of such tilings, given a set of tiles, is undecidable in general [GK72], and in [SMG+20] it was shown how to embed this problem into a PEPS tensor T of bond dimensions $D_1 = D_2 = D$ such that the associated state $|T_{n_1, n_2}\rangle$ on a $n_1 \times n_2$ periodic rectangular lattice is nonzero if and only if there exists a $n_1 \times n_2$ periodic tiling. The construction of such a tensor T is as follows. Let d be the number of tiles, label the tiles with an index $i \in [d]$, and similarly label the colors with an index $j \in [D]$. Then if the tile i has colors j_1, j_2, j_3, j_4 on respectively the left, right, upper and lower sides, define $T^{(i)} := |j_1\rangle \langle j_2| \otimes |j_3\rangle \langle j_4|$. It is not very hard to see that under this construction the resulting PEPS state $|T_{n_1, n_2}\rangle$ is nonzero if and only if there exists a $n_1 \times n_2$ periodic tiling. In fact, the argument in [SMG+20] is for PEPS tensors with boundary conditions, but the undecidability of the existence of periodic tilings [GK72] yields the same result for PEPS with periodic boundary conditions.

Interestingly, Proposition 3.3.13 shows that if one relaxes the problem to asking whether a PEPS tensor yields the zero state on any contraction graph, the problem is decidable, as we only have to check all graphs of size at most $\exp(O(D^2 \log D))$. Alternatively, the PEPS tensor yields the zero state on any contraction graph if and only if its minimal canonical form is the zero tensor. In the language of invariant theory, the PEPS tensor yields the zero state on any contraction graph if and only if it is in the null cone.

Example 3.3.17. The following is the smallest set of tiles that only gives aperiodic tilings, meaning that if we take any rectangle with periodic boundary conditions, the associated PEPS equals zero [JR21].



On the other hand it is easy to construct a geometry for which the associated PEPS is nonzero:



3. The minimal canonical form of a tensor network

In general, Proposition 3.3.13 together with the reduction in [SMG+20] shows that given a set of tiles with D colors, then there exists a ‘generalized tiling’ (i.e. an arbitrary way to glue together the edges of the tiles) on some closed (possibly non-orientable) surface if and only if such a generalized tiling exists using at most $\exp(O(D^2 \log D))$ tiles. The problem of deciding, given a set of tiles, whether there exists some generalized tiling is thus a decidable problem. The construction in Proposition 3.3.15 in fact used a PEPS corresponding to a tiling problem, showing that there are indeed situations where the smallest possible generalized tilings are of size at least $\exp(\Omega(D))$.

As argued in [SMG+20] their undecidability result excludes the possibility of a computable canonical form for two-dimensional PEPS which is such that two tensors T, S yield the same state on all periodic lattices (so $|T_{n_1, n_2}\rangle = |S_{n_1, n_2}\rangle$ for all n_1, n_2) if and only if they have the same canonical form. On the other hand, we saw in Corollary 3.3.9 that any two *normal* tensors which yield the same state on a periodic lattice are related by a local gauge transformation. However, even if generic tensors are normal, in two spatial dimensions many interesting tensors describing physical systems are not normal, in particular those associated to topological order, either conventional or symmetry-protected [CPSV21]. One way to interpret our Fundamental Theorem (Theorem 3.3.11) is that for some tensors it does not suffice to place them on periodic lattices and that the state they describe has a type of topological order which is only revealed by placing the states on a (possibly non-orientable) two-dimensional manifold other than a torus. This is an idea which is worth exploring in the future, and it is reminiscent of the well-known fact that different topological sectors can be detected by imposing different boundary conditions [CPSV21].

3.3.5. When does one need the orbit closure?

In general, finding the minimal canonical form requires one to go to the closure of the orbit of the action by the gauge group. In other words, if $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is a PEPS tensor in m spatial dimensions, then there may not exist a minimal canonical form T_{\min} of the form $(g_1, \dots, g_m) \cdot T$, but only one that can be written as a limit of such tensors: $T_{\min} = \lim_{j \rightarrow \infty} (g_1^{(j)}, \dots, g_m^{(j)}) \cdot T$. In other words, such a T is not *polystable* in the language of Section 2.3.4. When is taking limits really necessary? In this section we will discuss conditions under which one does not need to go to the closure and give an example where it is required. We consider PEPS tensors in m spatial dimensions, and fix bond dimensions D_1, \dots, D_m and physical dimension d . We denote by $G = \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$.

We will now argue that given a tensor $S \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ in minimal canonical form, if there exists a T which has S as a canonical form and which requires taking an orbit closure, then the tensor S must have a continuous symmetry. We formalize the notion of a continuous symmetry by a *multiplicative one-parameter subgroup* of G , which is a homomorphism of Lie groups $\phi: \mathbb{C}^\times \rightarrow G$. Given such a homomorphism we will write $\mathbf{g}(z)$ for $\phi(z)$ and we will say that $\mathbf{g}(z)$ is nontrivial if $\mathbf{g}(z)$ is not proportional to the identity for all $z \in \mathbb{C}^\times$.

The result we are aiming for is a consequence of the Hilbert–Mumford criterion in geometric invariant theory. If $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is any tensor, and T_{\min} is an

associated minimal canonical form, then $G \cdot T_{\min}$ is a closed orbit (by the Kempf–Ness Theorem, see Theorem 2.4.4). The Hilbert–Mumford criterion (Theorem 2.3.16) then implies that there exists a one-parameter subgroup $\mathfrak{g}(z) \in G$ such that

$$\lim_{z \rightarrow 0} \mathfrak{g}(z) \cdot T = S$$

where $S \in G \cdot T_{\min}$.

Proposition 3.3.18 (Non-closed implies symmetry). *Suppose $S \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is such that $G \cdot S$ is closed (in particular this is valid if S is in minimal canonical form). Suppose that there exists T such that $S \in \overline{G \cdot T}$ but $S \notin G \cdot T$, then there exists a nontrivial one-parameter subgroup $\mathfrak{g}(z) \subset G, z \in \mathbb{C}^\times$ such that $\mathfrak{g}(z) \cdot S = S$ for all $z \in \mathbb{C}^\times$.*

Proof. By the Hilbert–Mumford criterion there exists $\mathfrak{g} \in G$ and a one-parameter subgroup $\mathfrak{h}(z) \in G$ such that

$$\lim_{z \rightarrow 0} \mathfrak{h}(z) \cdot T = \mathfrak{g} \cdot S.$$

This one-parameter subgroup must be nontrivial since $S \notin G \cdot T$. Let $\mathfrak{g}(z) = \mathfrak{g}^{-1} \mathfrak{h}(z) \mathfrak{g}$. Then

$$\begin{aligned} \mathfrak{g}(z) \cdot S &= \mathfrak{g}^{-1} \mathfrak{h}(z) \mathfrak{g} \cdot S = \lim_{w \rightarrow 0} \mathfrak{g}^{-1} \mathfrak{h}(z) \mathfrak{h}(w) \cdot T \\ &= \lim_{w \rightarrow 0} \mathfrak{g}^{-1} \mathfrak{h}(zw) \cdot T \\ &= \mathfrak{g}^{-1} \cdot (\mathfrak{g} \cdot S) = S \end{aligned}$$

confirming that $\mathfrak{g}(z)$ is a symmetry for S . □

Example 3.3.19. Returning to the GHZ state in Example 3.2.4, we note that it indeed has a one-parameter subgroup symmetry, for instance for

$$\mathfrak{g}(z) = \begin{bmatrix} 1 & 0 \\ 0 & z \end{bmatrix}$$

it holds that $\mathfrak{g}(z) \cdot M = M$.

An important class of examples of PEPS tensors which lead to closed orbits are injective and normal tensors, already defined in Section 3.3.2. For those tensors (in particular for normal MPS) one does not need to take closures to construct the minimal canonical form. In fact we show that if there is any normal tensor in $\overline{G \cdot T}$, then $G \cdot T$ is closed (and in particular contains a minimal canonical form for T). A similar result has been shown for the case of MPS in [MGSC18] and has applications in the classification of two-dimensional SPT phases. This is a nice example where the geometric invariant theory framework allows for a particularly simple and conceptually elegant proof.

Proposition 3.3.20 (Canonical form normal PEPS). *Suppose $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ is such that its orbit closure $\overline{G \cdot T}$ contains a normal tensor. Then $\overline{G \cdot T} = G \cdot T$.*

3. The minimal canonical form of a tensor network

Proof. By Proposition 3.3.18 it suffices to show that if T is normal, then $G \cdot T$ is closed and there is no nontrivial one-parameter subgroup $\mathfrak{g}(z)$ such that $\mathfrak{g}(z) \cdot T = T$ for all $z \in \mathbb{C}^\times$.

Let \tilde{T} be the $n = n_1 \times \cdots \times n_m$ blocking of T such that \tilde{T} is injective. So, if we let $\tilde{D}_i = D_i^{n_1 \dots n_{i-1} n_{i+1} \dots n_m}$ and $\tilde{d} = d^n$,

$$\tilde{T} \in \text{Mat}_{\tilde{D}_1 \dots \tilde{D}_m \times \tilde{D}_1 \dots \tilde{D}_m}^{\tilde{d}}.$$

Let S be any tensor in $\overline{G \cdot T}$ and let \tilde{S} be the $n_1 \times \cdots \times n_m$ blocking of S . Since $S \in \overline{G \cdot T}$ there must be a sequence $\mathfrak{g}^{(j)} = (g_1^{(j)}, \dots, g_m^{(j)}) \in G$ for $j \in \mathbb{N}$ such that

$$\lim_{j \rightarrow \infty} \mathfrak{g}^{(j)} \cdot T = S.$$

Since $\mathfrak{g} \cdot T$ is invariant under rescaling the g_k by a constant, we may assume that $\|g_k^{(j)}\|_\infty = 1$ for all k and j . If we let $\tilde{g}_k^{(j)} = (g_k^{(j)})^{\otimes n_1 \dots n_{i-1} n_{i+1} \dots n_m}$ and $\tilde{\mathfrak{g}}^{(j)} = (\tilde{g}_1^{(j)}, \dots, \tilde{g}_m^{(j)})$ then

$$\lim_{j \rightarrow \infty} \tilde{\mathfrak{g}}^{(j)} \cdot \tilde{T} = \tilde{S}.$$

Now, interpret \tilde{T} as an element of $(\mathbb{C}^{\tilde{D}} \otimes \mathbb{C}^{\tilde{D}})^{\tilde{d}}$ where $\tilde{D} = \tilde{D}_1 \dots \tilde{D}_m$, so

$$\tilde{T} = (\tilde{T}^{(i)})_{i=1}^{\tilde{d}}, \quad \tilde{T}^{(i)} \in \mathbb{C}^{\tilde{D}} \otimes \mathbb{C}^{\tilde{D}}.$$

Then the fact that \tilde{T} is injective implies that there exists a tensor $\tilde{M} \in (\mathbb{C}^{\tilde{D}} \otimes \mathbb{C}^{\tilde{D}})^{\tilde{d}}$ which is an inverse to \tilde{T} in the sense that

$$\sum_{i=1}^{\tilde{d}} \tilde{T}^{(i)} (\tilde{M}^{(i)})^\dagger = I_{\tilde{D}^2}$$

is the identity map. Let $\tilde{N}^{(j)}$ be the contraction of $\tilde{\mathfrak{g}}^{(j)} \cdot \tilde{T}$ with \tilde{M} :

$$\tilde{N}^{(j)} = \sum_{i=1}^{\tilde{d}} \left(\tilde{\mathfrak{g}}^{(j)} \otimes (\tilde{\mathfrak{g}}^{(j)})^{-T} \tilde{T}^{(i)} \right) (\tilde{M}^{(i)})^\dagger$$

(writing $\tilde{\mathfrak{g}}^{(j)} = \tilde{g}_1^{(j)} \otimes \dots \otimes \tilde{g}_m^{(j)}$ in a slight abuse of notation). Then, $\tilde{N}^{(j)}$ must be a converging sequence (since $\tilde{\mathfrak{g}}^{(j)} \cdot \tilde{T}$ is so). On the other hand, since \tilde{M} is the inverse to \tilde{T} ,

$$\tilde{N}^{(j)} = \tilde{\mathfrak{g}}^{(j)} \otimes (\tilde{\mathfrak{g}}^{(j)})^{-T}.$$

The fact that this sequence converges implies that $\|(\tilde{\mathfrak{g}}^{(j)})^{-T}\|_\infty = \|(\tilde{\mathfrak{g}}^{(j)})^{-1}\|_\infty$ is bounded and hence there is some constant C such that for all $k \in [m]$ and $j \in \mathbb{N}$ we may bound $\|(g_k^{(j)})^{-1}\|_\infty \leq C$. However, this implies that $\mathfrak{g}^{(j)}$ is contained in a compact subset of G and therefore has a converging subsequence, which in turn implies that

$$S = \lim_{j \rightarrow \infty} \mathfrak{g}^{(j)} \cdot T \in G \cdot T.$$

So, we conclude that $G \cdot T$ is closed. Secondly, suppose that there exists a nontrivial one-parameter subgroup $\mathfrak{g}(z)$ such that $\mathfrak{g}(z) \cdot T = T$ for all $z \in \mathbb{C}^\times$. Using the same notation as before, this implies that there exists a one-parameter subgroup $\tilde{\mathfrak{g}}(z)$ such that $\tilde{\mathfrak{g}}(z) \cdot \tilde{T} = \tilde{T}$. However, applying the inverse \tilde{M} , this implies

$$\tilde{\mathfrak{g}}(z) \otimes \tilde{\mathfrak{g}}(z)^{-T} = I$$

which implies that $\mathfrak{g}(z)$ must be proportional to the identity for all $z \in \mathbb{C}^\times$. \square

Beyond normal PEPS states there are also other states of interest where Proposition 3.3.18 implies that one never needs to go to the closure to obtain the minimal canonical form.

Example 3.3.21. In two spatial dimensions an important example of a PEPS state which is not normal is the toric code. This is a state usually defined on a qubit lattice. To write it as a PEPS state one may group together four physical sites into a single site of four qubits. The toric code PEPS tensor is then given, as a map from the bond legs to the physical legs, by $T = \frac{1}{2}I^{\otimes 4} + \frac{1}{2}Z^{\otimes 4}$. Alternatively, for $i, j, k, l \in \{0, 1\}$

$$T^{(i,j,k,l)} = \begin{cases} |i\rangle \langle j| \otimes |l\rangle \langle k| & \text{if } i + j + k + l \text{ is even,} \\ 0 & \text{if } i + j + k + l \text{ is odd.} \end{cases}$$

This tensor is in minimal canonical form, since all virtual marginals are maximally mixed. We will now verify that this tensor has a finite symmetry group, and hence (as opposed to the GHZ state) there are no tensors for which T is in their orbit closure while not in the orbit itself. Suppose that $\mathfrak{g} \cdot T = T$ for $\mathfrak{g} = (g_1, g_2)$ with $g_k \in GL(2)$ for $k = 1, 2$. This is equivalent to

$$g_1 \otimes g_1^{-T} \otimes g_2 \otimes g_2^{-T} |i\rangle |j\rangle |k\rangle |l\rangle = |i\rangle |j\rangle |k\rangle |l\rangle.$$

for all $i + j + k + l = 0 \pmod{2}$. We can choose i and j arbitrary, so g_1 must be diagonal. By the same reasoning, g_2 must be diagonal as well. If we let

$$g_i = \begin{bmatrix} g_{i,0} & 0 \\ 0 & g_{i,1} \end{bmatrix}$$

then we find $g_{1,i}g_{2,k} = g_{1,j}g_{2,l}$ for all $i + j + k + l = 0 \pmod{2}$. By choosing $i \neq j$ and $k \neq l$ it is easy to see that this implies that after scaling by a global constant (which is irrelevant) $g_{i,j} \in \pm 1$ so we cannot have a nontrivial one-parameter subgroup symmetry.

Example 3.3.22. The previous example can be generalized to arbitrary quantum double models for abelian groups G . For an arbitrary finite group G we may construct a PEPS tensor (also known as a G -isometric PEPS tensor) as follows. The Hilbert space along each of the bond legs consists of the group algebra $\mathbb{C}[G]$ with basis $\{|g\rangle\}_{g \in G}$, so the bond dimension is $D = |G|$. The group G acts by the regular representation on $\mathbb{C}[G]$ as $g|h\rangle = |gh\rangle$. The physical Hilbert space is given by $\mathbb{C}[G]^{\otimes 4}$. Then the PEPS tensor is given, as a map from the bond Hilbert spaces to the physical Hilbert space as

$$T = \frac{1}{|G|} \sum_{g \in G} g \otimes \bar{g} \otimes g \otimes \bar{g}$$

3. The minimal canonical form of a tensor network

The toric code tensor is a special case of this construction for $G = \mathbb{Z}_2$. Essentially the same argument as for the toric code shows that (up to a global constant) the symmetries of this tensor form a discrete set if the group G is abelian and hence $\mathbb{C}[G]$ decomposes into one-dimensional irreducible representations. Therefore, $GL(D) \times GL(D) \cdot T = \overline{GL(D) \times GL(D)} \cdot T$.

Example 3.3.23. To give a nontrivial example where we do have a continuous symmetry, and we have non-closed orbits, we use a construction inspired by [DCS18], which investigates PEPS with continuous virtual symmetries. Consider a 2-dimensional PEPS tensor T with physical and bond dimensions all equal to two, given by

$$\begin{aligned} T^{(0)} &= \sum_{i,j \in \{0,1\}} |i\rangle \langle j| \otimes |j\rangle \langle i| \\ T^{(1)} &= \sum_{i,j \in \{0,1\}} |i\rangle \langle j| \otimes X|i\rangle \langle j|X. \end{aligned}$$

In the standard basis we may write this out as

$$T^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad T^{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

See [DCS18] for a graphical notation, expressing contractions as loop diagrams. All the virtual marginals of T are maximally mixed, so T is in minimal canonical form. It is now easy to see that $g(z) = (h(z), h(z))$ is a one-parameter subgroup symmetry for

$$h(z) = \begin{bmatrix} 1 & 0 \\ 0 & z \end{bmatrix}.$$

Indeed, since $h(z)|i\rangle \langle j| h(z)^{-1} = z^{i-j}|i\rangle \langle j|$ and $h(z)X|i\rangle \langle j|X h(z)^{-1} = z^{j-i}X|i\rangle \langle j|X$

$$\begin{aligned} (h(z) \otimes h(z)) T^{(0)} \left(h(z)^{-1} \otimes h(z)^{-1} \right) &= \sum_{i,j \in \{0,1\}} z^{i-j} |i\rangle \langle j| \otimes z^{j-i} |j\rangle \langle i| = T^{(0)} \\ (h(z) \otimes h(z)) T^{(1)} \left(h(z)^{-1} \otimes h(z)^{-1} \right) &= \sum_{i,j \in \{0,1\}} z^{i-j} |i\rangle \langle j| \otimes z^{j-i} X|i\rangle \langle j|X = T^{(1)}. \end{aligned}$$

Let us construct an explicit example where we need the closure to reach the minimal canonical form. Let $N = |1\rangle \langle 0| \otimes |1\rangle \langle 0|$ and let

$$S^{(0)} = T^{(0)} + N \quad \text{and} \quad S^{(1)} = T^{(1)} + N.$$

In the standard basis

$$S^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad S^{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Now, since $h(z) |1\rangle \langle 0| h(z)^{-1} = z |1\rangle \langle 0|$ and T is invariant under $g(z)$,

$$(h(z) \otimes h(z)) S^{(i)} \left(h(z)^{-1} \otimes h(z)^{-1} \right) = T^{(i)} + z^2 N$$

so

$$\lim_{z \rightarrow 0} g(z) \cdot S = T.$$

On the other hand, since $1 = \text{rank}(T^{(1)}) \neq \text{rank}(S^{(1)}) = 2$ we see that S is not in the orbit of T .

3.4. Algorithms for computing minimal canonical forms

In this section we address the question of how to compute minimal canonical forms algorithmically. We will discuss two algorithms (and sketch potential applications in Section 3.5). The first one is eminently practical and stated explicitly in Algorithm 3.1. The second one has a better runtime dependence in theory, but is less practical. We follow and apply the general framework of [BFG+19] but give some tighter bounds in our setting.

Before discussing our results and presenting our algorithm in more detail, we discuss what it means to compute a minimal canonical form. In general, minimal canonical forms cannot be represented exactly in finite precision, so one is naturally led to look for approximations. Then there are at least three natural choices of what it might mean to *approximately compute a minimal canonical form* of a given PEPS tensor T :

- *ℓ^2 -error in the space of tensors:* Given $\delta > 0$, find a tensor $S \in G \cdot T$ that is δ -close in ℓ^2 -norm to a minimal canonical form T_{\min} of T . It is natural consider relative error (but see Remark 3.4.13):

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \delta. \quad (3.4.1)$$

- *ℓ^2 -error in the first-order characterization:* Given $\varepsilon > 0$, find a tensor $S \in G \cdot T$ such that

$$\frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2} \leq \varepsilon \quad \text{where } \sigma = |S\rangle \langle S|. \quad (3.4.2)$$

- *error in the norm of the tensor:* Given $\zeta > 0$, find a tensor $S \in G \cdot T$ whose norm is almost minimal:

$$\frac{\|T_{\min}\|}{\|S\|} \geq 1 - \zeta. \quad (3.4.3)$$

Equation (3.4.3) corresponds to the norm minimization problem (Problem 2.6.3), whereas Eq. (3.4.2) corresponds to the scaling problem (Problem 2.6.4). We already know that Eq. (3.4.1) holds with $\delta = 0$ if and only if Eq. (3.4.2) holds with $\varepsilon = 0$

3. The minimal canonical form of a tensor network

if and only if Eq. (3.4.3) holds with $\zeta = 0$ (by Theorem 3.3.8 and the definition of the minimal canonical form). In Section 3.4.2 we will show that the three error measures can be related in a precise way. The quantitative relation between ε and ζ is a special case of the non-commutative duality theorem (Theorem 2.6.7). Their relation to Eq. (3.4.1) is new and relies on extending an argument from [KLLR18] in the setting of operator scaling; accordingly, we may target either.

3.4.1. First-order algorithm

We start by motivating our first algorithm, which we present explicitly in Algorithm 3.1, and recall several results from Chapter 2. Let $G = \text{GL}(D_1) \times \cdots \times \text{GL}(D_m)$ and $K = \text{U}(D_1) \times \cdots \times \text{U}(D_m)$. Suppose we are given a tensor $0 \neq T = (T^{(i)})_{i=1}^d \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ and we would like to approximately compute a minimal canonical form T_{\min} . Since the latter is defined as a minimum norm tensor in the orbit closure, a natural way to address this is by minimizing or “infimizing” the norm or, equivalently, the log-norm or Kempf–Ness function $F_T: G \rightarrow \mathbb{R}$ given by

$$F_T(\mathbf{g}) = \log \|\mathbf{g} \cdot T\| = \frac{1}{2} \log \|\mathbf{g} \cdot T\|^2.$$

Since $F_T(\mathbf{g}) = F_T(\mathbf{k}\mathbf{g})$ for all $\mathbf{k} \in K$ and $\mathbf{g} \in G$, the objective function F_T can be defined on the space $K \backslash G := \{\mathbf{K}\mathbf{g} : \mathbf{g} \in G\}$ of right K -cosets in the gauge group G . This space may be endowed with a natural Riemannian metric, yielding a simply-connected complete Riemannian manifold with *non-positive curvature* [BH13; Bha09]. In particular, between any two points there exist unique *geodesics* (here: shortest paths). Explicitly, the geodesics through $\mathbf{g} = (g_1, \dots, g_m) \in G$ take the form $\mathbf{K}(e^{tX_1}g_1, \dots, e^{tX_m}g_m)$ for $\mathbf{X} = (X_1, \dots, X_m) \in \mathbb{H}$.⁵

The point then is the following: While not convex in the ordinary sense, the function $F_T(\mathbf{p})$ is *geodesically convex*, that is, convex along these geodesics. This means for any $(g_1, \dots, g_m) \in G$ and $(X_1, \dots, X_m) \in \mathbb{H}$,

$$\partial_{t=0}^2 F_T(e^{tX_1}g_1, \dots, e^{tX_m}g_m) \geq 0.$$

Therefore, a reasonable approach to minimizing F_T is to use a gradient descent. Moreover, the computation done in Eq. (3.3.4) shows that the gradient at $\mathbf{g} = \mathbf{I} = (I_{D_1}, \dots, I_{D_m})$ is

$$\begin{aligned} \partial_{t=0} F_T(e^{tX_1}, \dots, e^{tX_m}) &= \frac{1}{2\|\mathbf{T}\|^2} \partial_{t=0} \|(e^{tX_1}, \dots, e^{tX_m}) \cdot \mathbf{T}\|^2 \\ &= \frac{1}{\text{Tr } \rho} \sum_{k=1}^m \text{Tr} \left[X_k \left(\rho_{k,1} - \rho_{k,2}^T \right) \right]. \end{aligned} \quad (3.4.4)$$

where $\rho = |\mathbf{T}\rangle \langle \mathbf{T}|$. Accordingly, starting at $\mathbf{g} = \mathbf{I}$ and moving along the geodesic with this direction, we should take a gradient step of the form

$$\mathbf{T} \mapsto \mathbf{g} \cdot \mathbf{T}, \quad \text{where } \mathbf{g} := \left(e^{-\eta \frac{1}{\text{Tr } \rho} (\rho_{1,1} - \rho_{1,2}^T)}, \dots, e^{-\eta \frac{1}{\text{Tr } \rho} (\rho_{m,1} - \rho_{m,2}^T)} \right),$$

⁵We can also identify $K \backslash G$ with $P = \text{PD}(D_1) \times \cdots \times \text{PD}(D_m)$ by the map $\mathbf{K}\mathbf{g} \mapsto \mathbf{g}^\dagger \mathbf{g}$. Then the geodesics can be written as $(\sqrt{p_1} e^{tY_1} \sqrt{p_1}, \dots, \sqrt{p_m} e^{tY_m} \sqrt{p_m})$, where $p_k = g_k^\dagger g_k$ and the Y_k are certain Hermitian matrices. These are tuples of the familiar geodesics of $\text{PD}(D_k)$, see e.g. Chapter 7 and Section 9.2 or [BH13; Bha09].

Algorithm 3.1: Computing PEPS normal forms

Input: A uniform PEPS tensor $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ and $\varepsilon > 0$.

Output: A gauge transformation $\mathbf{g} \in \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$.

```

1  $\mathbf{g}^{(0)} \leftarrow (I_{D_1}, \dots, I_{D_m});$ 
2 for  $t = 0, 1, \dots$  do
3    $T^{(t)} \leftarrow \mathbf{g}^{(t)} \cdot T;$ 
4    $\rho^{(t)} \leftarrow |T^{(t)}\rangle \langle T^{(t)}|;$ 
5   if  $\frac{1}{(\text{Tr } \rho^{(t)})^2} \sum_{k=1}^m \|\rho_{k,1}^{(t)} - (\rho_{k,2}^{(t)})^\top\|^2 \leq \varepsilon^2$  then
6     return  $\mathbf{g}^{(t)};$ 
7   end if
8   for  $k = 1, \dots, m$  do
9      $g_k^{(t+1)} \leftarrow e^{-\frac{1}{4m} \frac{1}{\text{Tr } \rho^{(t)}} (\rho_{k,1}^{(t)} - (\rho_{k,2}^{(t)})^\top)} g_k^{(t)};$ 
10  end for
11 end for
    
```

for some suitable step size $\eta > 0$. Note that, crucially, this amounts to acting by the gauge group, i.e., will automatically remain in the G -orbit!

Similarly to the ordinary gradient descent in Euclidean space, under suitable hypotheses on a geodesically convex objective one can provide a “safe” choice for the step size η . In the present case, the objective F_T is $4m$ -smooth along geodesics (as follows from Lemma 3.4.7 and Proposition 2.6.6): for every $\mathbf{g} = (g_1, \dots, g_m) \in G$ and $\mathbf{X} = (X_1, \dots, X_m) \in \mathbb{H}$, one has

$$\partial_{t=0}^2 F_T(e^{tX_1} g_1, \dots, e^{tX_m} g_m) \leq 4m \|\mathbf{X}\|^2,$$

where $\|\mathbf{X}\|^2 = \sum_{k=1}^m \|X_k\|^2$. For such functions, $\eta = \frac{1}{4m}$ is a suitable step size and this is what we use in Algorithm 3.1. Below, we give formal guarantees for the performance of the algorithm. We remark that Theorem 3.4.1 is a special case of [BFG+19, Thm. 4.2].

Theorem 3.4.1. *Let $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ be such that $T_{\min} \neq 0$ (for some and hence for any minimal canonical form), and let $\varepsilon > 0$. Then Algorithm 3.1 outputs a group element $\mathbf{g} \in \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies*

$$\frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^\top\|^2} \leq \varepsilon, \quad \text{where } \sigma = |S\rangle \langle S|,$$

within $O(\frac{m}{\varepsilon^2} \log \frac{\|T\|}{\|T_{\min}\|})$ iterations.

Proof. We analyze Algorithm 3.1: although we could appeal to a general result (Proposition 6.5.3), we give the analysis in this concrete setting. For $t = 0, 1, 2, \dots$ and $\mathbf{g}^{(t)}$ the group elements produced by the algorithm. If the algorithm does not terminate in the t -th iteration, then, using Eq. (3.4.4),

$$F_T(\mathbf{g}^{(t+1)}) - F_T(\mathbf{g}^{(t)})$$

3. The minimal canonical form of a tensor network

$$\begin{aligned}
&= F_{T^{(t)}}(e^{-\frac{1}{4m} \frac{1}{\text{Tr } \rho^{(t)}} (\rho_{1,1}^{(t)} - (\rho_{1,2}^{(t)})^T)}, \dots, e^{-\frac{1}{4m} \frac{1}{\text{Tr } \rho^{(t)}} (\rho_{m,1}^{(t)} - (\rho_{m,2}^{(t)})^T)}) - F_{T^{(t)}}(\mathbf{I}) \\
&= F_{T^{(t)}}(e^{-\frac{1}{4m} \nabla F_{T^{(t)}}(\mathbf{I})}) - F_{T^{(t)}}(\mathbf{I}) \\
&\leq \text{Tr} \left[\nabla F_{T^{(t)}}(\mathbf{I}) \cdot \left(-\frac{1}{4m} \nabla F_{T^{(t)}}(\mathbf{I}) \right) \right] + \frac{m}{8} \left\| -\frac{1}{m} \nabla F_{T^{(t)}}(\mathbf{I}) \right\|^2 \\
&= -\frac{1}{8m} \|\nabla F_{T^{(t)}}(\mathbf{I})\|^2 < -\frac{\varepsilon^2}{8m},
\end{aligned}$$

where the first inequality follows since F_T is a convex and $4m$ -smooth function (see Example 3.4.6 and Proposition 2.6.6). Accordingly, if the algorithm has not terminated up to and including the t -th iteration, then

$$\log \frac{\|T_{\min}\|}{\|T\|} \leq \log \|g^{(t)} \cdot T\| - \log \|T\| = F_T(g^{(t)}) - F_T(g^{(0)}) < -t \frac{\varepsilon^2}{8m},$$

or

$$t < \frac{8m}{\varepsilon^2} \log \frac{\|T\|}{\|T_{\min}\|}. \quad \square$$

The iteration bound of Theorem 3.4.1 involves $\|T_{\min}\|$. If the entries of T are given by some finite number of bits then this quantity can be estimated in an *a priori* fashion, by first rescaling T such that its entries are given by Gaussian integers, i.e., are in $\mathbb{Z}[i]$, and then using the following result.

Proposition 3.4.2. *Let $T \in \text{Mat}_{D_1 \cdots D_m \times D_1 \cdots D_m}^d$ with $T_{\min} \neq 0$, and assume that all entries of T are in $\mathbb{Z}[i]$. Then,*

$$\|T_{\min}\| \geq \frac{1}{\prod_{j=1}^m D_j}.$$

Proof. We use the fact that the invariant ring is generated by the functions $P_{\pi,i}$ defined in Eq. (3.3.3). Since $T_{\min} \neq 0$, there exist $n \geq 1$, $\pi \in S_n^m$ and $i \in [d]^n$ such that $P_{\pi,i}(T) \neq 0$. But $P_{\pi,i}$ is a polynomial with integer coefficients in the entries of T ; therefore, evaluating it on T with entries in $\mathbb{Z}[i]$ must yield $|P_{\pi,i}(T)| \geq 1$. Furthermore, it is an invariant under the PEPS action, so we deduce for any $g \in G$:

$$\begin{aligned}
1 &\leq |P_{\pi,i}(T)| = |P_{\pi,i}(g \cdot T)| = \left| \text{Tr} \left[(R_{\pi_1} \otimes \cdots \otimes R_{\pi_m}) ((g \cdot T^{(i_1)}) \otimes \cdots \otimes (g \cdot T^{(i_n)})) \right] \right| \\
&\leq \|R_{\pi_1} \otimes \cdots \otimes R_{\pi_m}\| \cdot \|(g \cdot T^{(i_1)}) \otimes \cdots \otimes (g \cdot T^{(i_n)})\|.
\end{aligned}$$

Since each R_{π_j} is unitary, the same is true of their tensor product. As it acts on a space of dimension $(\prod_{j=1}^m D_j^2)^n$, one obtains

$$\|R_{\pi_1} \otimes \cdots \otimes R_{\pi_m}\| = \left(\prod_{j=1}^m D_j \right)^n.$$

Furthermore,

$$\|(g \cdot T^{(i_1)}) \otimes \cdots \otimes (g \cdot T^{(i_n)})\| \leq (\max_i \|g \cdot T^{(i)}\|)^n \leq \|g \cdot T\|^n.$$

Combining the two estimates, taking n -th roots and the infimum over $g \in G$ yields the desired estimate. \square

The above approach of evaluating an invariant to prove norm lower bounds is used in other settings as well, e.g., for tensor scaling in [BGO+18, Thm. 7.12], and for much more general actions in [BFG+19, Cor. 7.19]; but appealing to the latter result would result in a worse bound.

We obtain the following corollary, which implies an $\text{poly}(\frac{1}{\varepsilon}, \text{input size})$ -time algorithm, cf. [BFG+19, Rem. 8.1]:

Corollary 3.4.3. *Let $T \in \text{Mat}_{D_1 \cdots D_m \times D_1 \cdots D_m}^d$ be a tensor such that $T_{\min} \neq 0$ (for some and hence for any minimal canonical form). Assume that the entries of T are in $\mathbb{Q}[i]$ and given by storing the numerators and denominators in binary. Let $\varepsilon > 0$. Then Algorithm 3.1 outputs a group element $\mathbf{g} \in \text{GL}(D_1) \times \cdots \times \text{GL}(D_m)$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies*

$$\frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2} \leq \varepsilon, \quad \text{where } \sigma = |S\rangle \langle S|.$$

within $O(\frac{1}{\varepsilon^2} \cdot \text{poly}(\langle T \rangle))$ iterations, where $\langle T \rangle$ denotes the total number of bits used to represent T .

3.4.2. Relation between approximation errors

In Section 3.4.1, we discussed three natural notions of approximation error in Eqs. (3.4.1) to (3.4.3), and we gave an algorithm targeting Eq. (3.4.2), i.e., given a tensor T and $\varepsilon > 0$, we discussed how to obtain a tensor $S \in G \cdot T$ such that

$$\frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2} \leq \varepsilon \quad \text{where } \sigma = |S\rangle \langle S|.$$

We will now see that there is a precise quantitative relationship between these notions. As we will see, the following quantity will play a crucial role.

Definition 3.4.4. Given bond dimensions D_1, \dots, D_m , define

$$\gamma := \gamma(D_1, \dots, D_m) := \begin{cases} 1 & \text{if } m = 1, \\ D_1^{3/2} & \text{if } m = 2, \\ \frac{1}{\sum_{i=1}^m D_i} \cdot \frac{1}{(2m)^{(\sum_{i=1}^m D_i - 1)/2}} & \text{if } m \geq 3. \end{cases}$$

Note that γ is only inverse polynomially small in the bond dimension for $m = 1$, while it is exponentially small for $m \geq 2$. Then we have the following relation between Eqs. (3.4.2) and (3.4.3).

Theorem 3.4.5. *Let $0 \neq T \in \text{Mat}_{D_1 \cdots D_m \times D_1 \cdots D_m}^d$ and $S \in G \cdot T$. Then:*

$$1 - \frac{\varepsilon}{\gamma} \leq \frac{\|T_{\min}\|^2}{\|S\|^2} \leq 1 - \frac{\varepsilon^2}{8m} \quad \text{for} \quad \varepsilon := \frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2},$$

where $\sigma = |S\rangle \langle S|$ and γ is the constant defined in Definition 3.4.4. In particular, if $\varepsilon < \gamma$, then $T_{\min} \neq 0$.

3. The minimal canonical form of a tensor network

We will prove Theorem 3.4.5 by appealing to a non-commutative duality theorem stated in [BFG+19, Thm. 1.17], which we stated previously in Theorem 2.6.7. To apply this theorem in our setting, we must lower bound the *weight margin* as defined in Definition 2.6.5, a complexity measure defined by combinatorial data associated with representations. The parameter γ which appears in Definition 3.4.4 is a lower bound on this weight margin.

We briefly recall how to compute the weights for this particular setting:

Example 3.4.6. Let $GL(D)$ act on $\text{Mat}_{D \times D}$ by conjugation. A maximal subtorus of $GL(D)$ is given by the set $T(D) := (\mathbb{C}^\times)^D$ consisting of invertible diagonal $D \times D$ matrices, and its Lie algebra $\text{Lie}(T(D))$ consists of all diagonal matrices, which may be identified with \mathbb{C}^D . Then for $Y \in \mathbb{C}^D$, we have

$$e^{\text{diag}(Y)} E_{ij} e^{-\text{diag}(Y)} = e^{Y_i - Y_j} E_{ij},$$

where E_{ij} are the elementary matrices. Therefore the weights are given by the functionals $\omega_{ij}(Y) = Y_i - Y_j$, with corresponding weight spaces $V_{\omega_{ij}} = \mathbb{C} E_{ij}$. Note that ω_{ij} can be identified with $e_i - e_j \in \mathbb{C}^D$. The action of $GL(D)$ on $\text{Mat}_{D \times D}^d$ has the same weights, but now each weight space is d -dimensional.

Now consider the action of the gauge group $G = GL(D_1) \times \cdots \times GL(D_m)$ on $V = \text{Mat}_{D_1 \cdots D_m \times D_1 \cdots D_m}^d$, the space of PEPS tensors, as defined in Definition 3.3.3. As mentioned, a maximal torus for G is given by $T_G = T(D_1) \times \cdots \times T(D_m)$, and the Lie algebra of T_G may be identified with $\mathbb{C}^{D_1} \oplus \cdots \oplus \mathbb{C}^{D_m}$. Then it is easy to show that the weights are just tuples of weights as above, i.e.,

$$(e_{i_1} - e_{j_1}, \dots, e_{i_m} - e_{j_m})$$

with $i_k, j_k \in [D_k]$ for $k \in [m]$.

To prove Theorem 3.4.5, we still need to bound the parameters $\gamma(\pi)$ and $N(\pi)$ for our specific representations.

Lemma 3.4.7. *For the action of $G = GL(D_1) \times \cdots \times GL(D_m)$ on $V = \text{Mat}_{D_1 \cdots D_m \times D_1 \cdots D_m}^d$, the weight norm $N(\pi)$ is given by*

$$N(\pi) = \sqrt{2m},$$

and the weight margin $\gamma(\pi)$ is lower bounded as

$$\gamma(\pi) \geq \gamma,$$

where γ is the constant defined in Definition 3.4.4.

Proof. The expression for the weight norm follows directly from Example 3.4.6.

For $m = 1$, the lower bound on the weight margin follows from [BFG+19, Thm. 6.21]: the representation is a quiver representation, where the quiver is given by one vertex with d self-loops. For $m \geq 2$, the lower bound on the weight margin follows from [BFG+19, Thm. 6.10]. \square

Proof of Theorem 3.4.5. This follows by combining Theorem 2.6.7 and Lemma 3.4.7. \square

Now that we know that Eqs. (3.4.2) and (3.4.3) can be related to each other, we will relate these to Eq. (3.4.1). In the one direction, it is clear that Eq. (3.4.1) implies a small error in the sense of Eq. (3.4.3):

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \delta \quad \Rightarrow \quad \frac{\|T_{\min}\|}{\|S\|} \geq 1 - \frac{\|T_{\min} - S\|}{\|S\|} \geq 1 - \delta$$

In the remainder of this section we show that Eq. (3.4.2) implies a small error in the sense of Eq. (3.4.1), closing the circle. It is useful to make the following abbreviation for the gradient of the norm square function at the identity:

$$\tilde{\mu}(S) := \left(\sigma_{k,1} - \sigma_{k,2}^T \right)_{k=1}^m \in \text{Herm}(D_1) \oplus \cdots \oplus \text{Herm}(D_m), \quad \text{where } \sigma := |S\rangle \langle S|.$$

We write $\tilde{\mu}$ and not μ to distinguish it from the gradient of the log-norm, as in Eq. (3.4.4) and Definition 2.5.1, but note that

$$\|\tilde{\mu}(S)\| = \varepsilon \text{Tr}(\sigma) = \varepsilon \|S\|^2, \quad \text{where } \varepsilon = \frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2}. \quad (3.4.5)$$

Then we will consider the gradient flow of $\|\tilde{\mu}(S)\|^2 := \sum_{k=1}^m \|\rho_{k,1} - \rho_{k,2}^T\|^2$:

$$\begin{cases} S'(t) = -\nabla \|\tilde{\mu}\|^2(S(t)) \\ S(0) = S \end{cases} \quad (3.4.6)$$

We will see that the solution $S(t)$ to this ODE remains in the gauge orbit of S and that it converges to a minimal canonical form S_{\min} whose distance to S in the sense of Eq. (3.4.1) can be controlled using Eq. (3.4.2). We note here that the study of the gradient flow for the norm square of the moment map is an important tool in this area; see Section 12.2.2 for a more detailed discussion. While the following arguments work in complete generality, here we restrict to the gauge action of $G = \text{GL}(D_1) \times \cdots \times \text{GL}(D_m)$ since this is all we need.

We start by analyzing Eq. (3.4.6). Existence and uniqueness of the solution $S(t)$ of this ordinary differential equation on some maximal (possibly infinite) interval of definition $[0, t_{\max})$, where $t_{\max} \in (0, \infty]$, follows from Picard–Lindelöf theory. Then one can prove the following lemma, cf. [BFG+19, Prop. 3.27 and its proof]:

Lemma 3.4.8. *Let $S(t)$ be the solution to the dynamical system (3.4.6). Then, for all $t \in [0, t_{\max})$, we have*

$$(i) \quad \partial_t \|\tilde{\mu}(S(t))\|^2 = -\|S'(t)\|^2.$$

$$(ii) \quad \partial_t \|S(t)\|^2 = -8\|\tilde{\mu}(S(t))\|^2.$$

(iii) $S(t) \in G \cdot S$, i.e., the solution remains in the G -orbit of S at all times.

Proof. The first claim holds for any gradient flow.

Next, we note that, for all $Y \in \text{Herm}(D_1) \oplus \cdots \oplus \text{Herm}(D_m)$,

$$\langle \tilde{\mu}(S), Y \rangle = \frac{1}{2} \partial_{t=0} \|(e^{tY_1}, \dots, e^{tY_m}) \cdot S\|^2 = \langle S, \Pi(Y)S \rangle, \quad (3.4.7)$$

3. The minimal canonical form of a tensor network

where $\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{k=1}^m \text{Tr}[X_k Y_k]$ and we denote by $\Pi(\mathbf{Y})$ the Lie algebra action of \mathbf{Y} , which is defined by

$$\Pi(\mathbf{Y})S := \partial_{t=0} \left((e^{tY_1}, \dots, e^{tY_m}) \cdot S \right).$$

By differentiating Eq. (3.4.7) with respect to S in some direction $W \in V$ (an operation we denote by D_W),

$$\langle D_W \tilde{\mu}(S), \mathbf{Y} \rangle = \langle W, \Pi(\mathbf{Y})S \rangle + \langle S, \Pi(\mathbf{Y})W \rangle = 2 \text{Re} \langle W, \Pi(\mathbf{Y})S \rangle.$$

Accordingly, for all $W \in V$,

$$D_W \|\tilde{\mu}(S)\|^2 = 2 \langle D_W \tilde{\mu}(S), \tilde{\mu}(S) \rangle = 4 \text{Re} \langle W, \Pi(\tilde{\mu}(S))S \rangle.$$

Thus we have proved that the gradient of $\|\tilde{\mu}\|^2$ is given by the following clean formula:

$$\nabla \|\tilde{\mu}\|^2(S) = 4\Pi(\tilde{\mu}(S))S. \quad (3.4.8)$$

The second item follows from this and Eq. (3.4.7),

$$\begin{aligned} \partial_t \|S(t)\|^2 &= 2 \langle S(t), S'(t) \rangle = -2 \langle S(t), \nabla \|\tilde{\mu}\|^2(S(t)) \rangle \\ &= -8 \langle S(t), \Pi(\tilde{\mu}(S(t)))S(t) \rangle = -8 \|\tilde{\mu}(S(t))\|^2. \end{aligned}$$

As Eq. (3.4.8) states that $S'(t)$ is a tangent vector of the G -orbit through $S(t)$, the third item also follows. \square

Using the preceding, the following key lemma shows that if $S_{\min} \neq 0$ then $\tilde{\mu}(S(t)) \rightarrow 0$ sufficiently quickly, without $S(t)$ moving too much. Our argument follows [KLLR18], which treats the case $m = 1$.

Lemma 3.4.9. *Let $S(t)$ denote the solution of Eq. (3.4.6) for a tensor $S(0) = S$ with $S_{\min} \neq 0$ (for some and hence for any minimal canonical form). Consider any τ such that $\tilde{\mu}(S(\tau)) \neq 0$. Then there exists*

$$\tau' \leq \tau + \frac{1}{4\gamma \|\tilde{\mu}(S(\tau))\|}.$$

such that

$$\|\tilde{\mu}(S(\tau'))\|^2 = \frac{\|\tilde{\mu}(S(\tau))\|^2}{2} \quad (3.4.9)$$

(in fact τ' is the first time such that this is true) and, moreover,

$$\|S(\tau') - S(\tau)\| \leq \frac{1}{2\sqrt{2}} \sqrt{\frac{\|\tilde{\mu}(S(\tau))\|}{\gamma}}, \quad (3.4.10)$$

where γ is the constant from Definition 3.4.4.

Proof. Suppose that $\tau' > \tau$ is such that

$$\|\tilde{\mu}(S(\tau'))\|^2 > \frac{\|\tilde{\mu}(S(\tau))\|^2}{2}. \quad (3.4.11)$$

By Item (i) of Lemma 3.4.8,

$$\|\tilde{\mu}(S(t))\|^2 > \frac{\|\tilde{\mu}(S(\tau))\|^2}{2} \quad \forall t \in [\tau, \tau']$$

and hence, by Item (ii) of the same lemma,

$$\partial_t \|S(t)\|^2 = -8\|\tilde{\mu}(S(t))\|^2 < -4\|\tilde{\mu}(S(\tau))\|^2 \quad \forall t \in [\tau, \tau'].$$

Accordingly,

$$\|S(\tau')\|^2 - \|S(\tau)\|^2 < -4(\tau' - \tau)\|\tilde{\mu}(S(\tau))\|^2.$$

On the other hand, using the lower bound in Theorem 3.4.5 and Eq. (3.4.5),

$$\begin{aligned} \|S(\tau')\|^2 - \|S(\tau)\|^2 &\geq \|S(\tau')_{\min}\|^2 - \|S(\tau)\|^2 = \|S(\tau)_{\min}\|^2 - \|S(\tau)\|^2 \\ &= \|S(\tau)\|^2 \left(\frac{\|S(\tau)_{\min}\|^2}{\|S(\tau)\|^2} - 1 \right) \geq -\frac{\|\tilde{\mu}(S(\tau))\|}{\gamma}. \end{aligned}$$

Together, we find that for any τ' such that Eq. (3.4.11) holds, we must have

$$\tau' < \tau + \frac{1}{4\gamma\|\tilde{\mu}(S(\tau))\|}.$$

Accordingly, there must exist some minimal

$$\tau' \leq \tau + \frac{1}{4\gamma\|\tilde{\mu}(S(\tau))\|}. \quad (3.4.12)$$

such that

$$\|\tilde{\mu}(S(\tau'))\|^2 = \frac{\|\tilde{\mu}(S(\tau))\|^2}{2}. \quad (3.4.13)$$

Moreover, for this τ' we have

$$\begin{aligned} \|S(\tau') - S(\tau)\| &\leq \int_{\tau}^{\tau'} \|S'(t)\| dt = \int_{\tau}^{\tau'} \sqrt{-\partial_t \|\tilde{\mu}(S(t))\|^2} dt \\ &\leq \sqrt{\int_{\tau}^{\tau'} -\partial_t \|\tilde{\mu}(S(t))\|^2 dt} \sqrt{\int_{\tau}^{\tau'} 1 dt} \\ &= \sqrt{\|\tilde{\mu}(S(\tau))\|^2 - \|\tilde{\mu}(S(\tau'))\|^2} \sqrt{\tau' - \tau} \\ &\leq \frac{\|\tilde{\mu}(S(\tau))\|}{\sqrt{2}} \sqrt{\frac{1}{4\gamma\|\tilde{\mu}(S(\tau))\|}} \\ &= \frac{1}{2\sqrt{2}} \sqrt{\frac{\|\tilde{\mu}(S(\tau))\|}{\gamma}}, \end{aligned}$$

where we used the triangle inequality, then Item (i) of Lemma 3.4.8, then the Cauchy-Schwarz inequality, and finally Eqs. (3.4.12) and (3.4.13). \square

3. The minimal canonical form of a tensor network

We now prove the desired relation between Eqs. (3.4.1) and (3.4.2):

Theorem 3.4.10. *Let T be a tensor with $T_{\min} \neq 0$ (for some and hence for any minimal canonical form) and let $S \in G \cdot T$. Then there exists a minimal canonical form $T_{\min} \in \overline{G \cdot T}$ such that*

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \sqrt{\frac{2\varepsilon}{\gamma}} \quad \text{for} \quad \varepsilon := \frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2},$$

where γ is the constant from Definition 3.4.4.

Proof. If $\mu(S) = 0$ then $T_{\min} = S$ is a minimal canonical form of T and there is nothing to prove. Otherwise let us, for every $k \geq 0$, denote by τ_k the first time when

$$\|\tilde{\mu}(S(\tau_k))\|^2 = \frac{1}{2^k} \|\tilde{\mu}(S)\|^2,$$

so $\tau_0 = 0$. By Lemma 3.4.9,

$$\tau_k = \sum_{l=1}^k (\tau_l - \tau_{l-1}) \leq \sum_{l=1}^k \frac{1}{4\gamma \|\tilde{\mu}(S(\tau_{l-1}))\|} = \frac{1}{4\gamma} \sum_{l=1}^k \frac{1}{\sqrt{\frac{1}{2^{l-1}} \|\tilde{\mu}(S)\|}} \leq \frac{2^{k/2}}{\gamma \|\tilde{\mu}(S)\|}$$

In particular, $\tilde{\mu}(S(t)) \rightarrow 0$ as $t \rightarrow \infty$, since we know from Item (i) of Lemma 3.4.8 that $\|\tilde{\mu}(S(t))\|^2$ is monotonically decreasing.

Next, we prove that the subsequence $S(\tau_k)$ converges to a minimal canonical form of T with the desired properties. We first show that the $S(\tau_k)$ form a Cauchy sequence. Indeed, for any $k \leq l$, using Lemma 3.4.9,

$$\begin{aligned} \|S(\tau_k) - S(\tau_l)\| &\leq \sum_{m=k+1}^l \|S(\tau_m) - S(\tau_{m-1})\| \leq \sum_{m=k+1}^l \frac{1}{2\sqrt{2}} \sqrt{\frac{\|\tilde{\mu}(S(\tau_{m-1}))\|}{\gamma}} \\ &= \frac{1}{2\sqrt{2}} \sqrt{\frac{\|\tilde{\mu}(S)\|}{\gamma}} \sum_{m=k+1}^l \sqrt{\frac{1}{2^{m-1}}} \leq \sqrt{\frac{2\|\tilde{\mu}(S)\|}{\gamma}} \sqrt{\frac{1}{2^k}}, \end{aligned}$$

which shows that indeed $S(\tau_k)$ is a Cauchy sequence. If we denote by S' its limit, then $T' \in \overline{G \cdot S} = \overline{G \cdot T}$ (by Item (iii) of Lemma 3.4.8) and hence $T' \neq 0$ (since $T_{\min} \neq 0$ by assumption). Moreover, $\tilde{\mu}(T') = 0$ by the above, hence T' is a minimal canonical form of T . Finally,

$$\|S - T'\| = \lim_{l \rightarrow \infty} \|S(\tau_0) - S(\tau_l)\| \leq \sqrt{\frac{2\|\tilde{\mu}(S)\|}{\gamma}} = \|S\| \sqrt{\frac{2\varepsilon}{\gamma}}$$

using the preceding estimate and Eq. (3.4.5) □

By combining Corollary 3.4.3 and Theorem 3.4.10 it follows that using the first-order algorithm in Algorithm 3.1 with $\varepsilon := \gamma\delta^2/2$, in time $\text{poly}(\frac{1}{\gamma}, \frac{1}{\delta}, \text{input size})$ one can obtain a group element $\mathbf{g} \in G$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies Eq. (3.4.1), i.e.,

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \delta.$$

In the next section we will see that the dependence on δ can be improved to $\log(1/\delta)$, see Corollary 3.4.12.

3.4.3. Second-order algorithm

As promised earlier, there is also a second numerical method that one can use to approximate normal forms in our setting. This is a more sophisticated *second-order* method, which uses information about the Hessian of the Kempf–Ness function F_v to determine the direction in which to move (as is done for instance in Newton’s method), whereas the first-order method discussed in Section 3.4.1 and Algorithm 3.1 before only use information about the gradient (i.e., the moment map).

We rely on the box-constrained Newton method from [BFG+19, Algo. 5.2], which uses Newton steps constrained to a constant-sized box to make progress in the objective. It naturally minimizes the norm of the resulting vector, as opposed to the size of the gradient. We note that we could also appeal to the interior-point methods in Chapter 7, but the result is essentially the same. Its guarantees applied to our setting are as follows:

Theorem 3.4.11 ([BFG+19, Thm. 8.12]). *Let $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ be a tensor such that $T_{\min} \neq 0$ (for some and hence for any minimal canonical form). Assume that the entries of T are in $\mathbb{Q}[i]$ and given by storing the numerators and denominators in binary. Then there exists an algorithm that, given T and $0 < \zeta < 1$, returns a group element $\mathbf{g} \in \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies $\|S\| \leq \|T\|$ and*

$$\log \frac{\|S\|}{\|T_{\min}\|} \leq \zeta \quad \text{and hence} \quad \frac{\|T_{\min}\|}{\|S\|} \geq 1 - \zeta$$

in time $\text{poly}(\gamma^{-1}, D_1, \dots, D_m, \log(1/\zeta), \langle T \rangle)$, where γ is defined in Definition 3.4.4, and $\langle T \rangle$ is the total number of bits used to represent T .

By combining Theorem 3.4.11 with the results of Section 3.4.2, we arrive at the following result which was stated informally as Result 3 in the introduction.

Corollary 3.4.12. *Let $T \in \text{Mat}_{D_1 \dots D_m \times D_1 \dots D_m}^d$ be a tensor such that $T_{\min} \neq 0$ (for some and hence for any minimal canonical form). Assume that the entries of T are in $\mathbb{Q}[i]$ and given by storing the numerators and denominators in binary. Then there exists an algorithm that, given T and $0 < \delta < 1$, returns a group element $\mathbf{g} \in \text{GL}(D_1) \times \dots \times \text{GL}(D_m)$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies $\|S\| \leq \|T\|$ and*

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \delta,$$

in time $\text{poly}(\gamma^{-1}, D_1, \dots, D_m, \log(1/\delta), \langle T \rangle)$, where γ is defined in Definition 3.4.4, and $\langle T \rangle$ is the total number of bits used to represent T .

Proof. Apply the algorithm of Theorem 3.4.11 with

$$\zeta := \frac{\gamma^2}{64m} \delta^4 \tag{3.4.14}$$

to obtain in the stated runtime a group element $\mathbf{g} \in G$ such that the tensor $S := \mathbf{g} \cdot T$ satisfies $\|S\| \leq \|T\|$ and

$$\frac{\|T_{\min}\|}{\|S\|} \geq 1 - \zeta \quad \text{and hence} \quad \frac{\|T_{\min}\|^2}{\|S\|^2} \geq 1 - 2\zeta. \tag{3.4.15}$$

3. The minimal canonical form of a tensor network

We now check that S satisfies the desired condition. First, by Theorem 3.4.5, for $\sigma = |S\rangle\langle S|$ we have that

$$\frac{\|T_{\min}\|^2}{\|S\|^2} \leq 1 - \frac{\varepsilon^2}{8m} \quad \text{for} \quad \varepsilon := \frac{1}{\text{Tr } \sigma} \sqrt{\sum_{k=1}^m \|\sigma_{k,1} - \sigma_{k,2}^T\|^2},$$

and hence, using Eq. (3.4.15),

$$\varepsilon \leq \sqrt{8m \left(1 - \frac{\|T_{\min}\|^2}{\|S\|^2}\right)} \leq 4\sqrt{m\zeta}. \quad (3.4.16)$$

Finally, Theorem 3.4.10 implies that

$$\frac{\|S - T_{\min}\|}{\|S\|} \leq \sqrt{\frac{2\varepsilon}{\gamma}} \quad \text{and hence} \quad \frac{\|S - T_{\min}\|}{\|S\|} \leq \sqrt{\frac{2\varepsilon}{\gamma}} \leq \sqrt{\frac{8\sqrt{m\zeta}}{\gamma}} \leq \delta,$$

where used Eq. (3.4.16) and our choice of ζ in Eq. (3.4.14). This concludes the proof. \square

Remark 3.4.13. While Corollary 3.4.12 uses relative ℓ^2 -error, which is most natural, we can also obtain a guarantee in absolute error, say

$$\|S - T_{\min}\| \leq \delta',$$

by applying Corollary 3.4.12 with $\delta < \min(1, \delta'/\|T\|)$. As the second-order algorithm scales polynomially in $\log(1/\delta)$, this runs in time $\text{poly}(\gamma^{-1}, D_1, \dots, D_m, \log(1/\delta'), \langle T \rangle)$.

3.5. Conclusion and outlook

The current work is a theoretical one, proposing a new canonical form and proving some of its key properties. The fact that the minimal canonical form is rigorous in the sense that it can be proven to always exist as well as satisfy the basic properties discussed in Section 3.3 sets it apart from other heuristic approaches [PMV15; Eve18]. Besides this, we hope that the minimal canonical form will be of practical use in tensor network algorithms. Below we outline four potential directions for application. Detailed numerical study will be required to confirm the usefulness of these suggestions.

- (i) **Truncation of bond dimensions.** In many tensor network algorithms *truncation* of the bond dimension is a crucial step. This is especially the case for ground state finding algorithms based on imaginary time evolution (Time Evolving Block Decimation, TEBD) in which each step consist of applying an operator to the tensor network which increases the ground state approximation accuracy but also the bond dimension, and then truncating the bond dimension. One is given a tensor T with a certain bond dimension D , and one would like to find a tensor T' with a prescribed bond dimension $D' < D$ such that the tensor network state using T is approximated as accurately as possible by the tensor network state using T' . In one spatial

dimension, for MPS, there is a natural way to do this using canonical forms. For instance, one may use the left canonical form, in which case the reduced state ρ_2 on the right virtual dimension is maximally mixed. Then one truncates to the subspace spanned by the eigenvectors of the D' largest eigenvalues of the reduced state ρ_1 on the left virtual dimension.

The bond dimension truncation scheme for MPS is both computationally efficient and gives an optimal approximation given a prescribed bond dimension. For two-dimensional PEPS there is no truncation scheme known which has both these desirable properties, which is closely related to the lacking of the equivalent of a left or right canonical form. Various methods exist [LCB14b; JWX08], see for instance [RTP+20] for an overview of different methods. While these methods perform well in practice, in most cases good theoretical understanding is lacking. Here, we propose the following natural truncation scheme: given a tensor T , compute its minimal canonical form S . Then truncate to the subspace spanned by the eigenvectors corresponding to the D' largest eigenvalues.

This proposal leads various questions which should be addressed in follow-up work. First of all, it would be interesting to use such a truncation method in existing PEPS algorithms and study the performance of such schemes numerically. Secondly, as our methods are designed for uniform (translation-invariant) systems one would hope that they are also of use to iPEPS methods, where precisely the absence of a canonical form has led to heuristic approaches to gauge-fixing [PBT+15; PMV15] which work well in practice. We would like to emphasize that especially the (non-rigorously defined) canonical form in [PMV15] is fairly close in spirit to the minimal canonical form: it is defined by a condition similar (but different) to the characterization in Theorem 3.3.8. This canonical form has been shown to indeed improve convergence of imaginary time evolution algorithms, which offers some hope for the prospect of using the minimal canonical form for truncation purposes. Finally, a potential advantage of truncation schemes based on the minimal canonical form is that one could attempt to the framework of geometric invariant theory to prove that such a truncation scheme has good theoretical properties.

- (ii) **Numerical stability.** Using minimal canonical forms in variational algorithms may be helpful, since appropriate gauge fixing is known to enhance the stability of variational algorithms [LCB14a; PBT+15].
- (iii) **Boundary-based approaches.** PEPS have a very useful and explicit bulk-boundary correspondence [CPSV11], which allows one to map bulk properties in a region R to properties of the associated boundary state ρ_R , defined essentially as the reduced density matrix in the virtual indices of the PEPS tensor $|T_R\rangle$ obtained after blocking the original PEPS tensor T in the given region R . The key insight of [CPSV11], formalized later in [KLP19; PP23], is that if one interprets ρ_R as a Gibbs state $\rho_R = e^{-H_E}$, the properties of H_E (the so-called *entanglement Hamiltonian*) encode the properties of the bulk of the system. This has led for instance to new numerical methods to detect topological phase transitions [SPCP13]. Since H_E and ρ_R live in the virtual

3. The minimal canonical form of a tensor network

Hilbert space, it is crucial for this approach to be meaningful that the only gauge freedom one considers comes from unitaries, which do not change any of the relevant properties of H_E or ρ_R , rather than arbitrary invertible matrices. This is precisely what is ensured by working with the minimal canonical form.

- (iv) **Privacy in PEPS-based machine learning algorithms.** Tensor networks, and PEPS and MPS in particular, have been used as variational Ansätze in machine learning contexts [SS16; CPZ+17]. This has the appeal that one can import known optimization techniques in condensed matter problems to machine learning. Another potential advantage, compared to neural network-based approaches, lies in a higher interpretability; it is precisely the characterization of global properties in the local tensors of a tensor network which explains its success in quantum many-body problems. In [PHM+22] a new potential advantage of tensor networks in a machine learning context has been proposed, which we will now explain briefly. There are two possible ways to look at a trained neural network or tensor network: as a *black box* in which one has only access to the input-output relation or as a *white box* in which all internal parameters are provided. It is shown in [PHM+22], with machines trained in real data bases with medical records, that those internal parameters can reveal sensitive information from the training data set which however are not contained in the black-box picture. This white-box versus black-box scenario is the underlying problem behind obfuscation protocols⁶ and it is well known there that the perfect solution comes from the existence of a well-defined canonical form for the white-boxes that maps them one-to-one to the set of black boxes. The basic idea in [PHM+22] is that this can be done in MPS by defining a new canonical form which selects analytically and uniquely an element for each orbit of a normal MPS. However, as it is also discussed in [PHM+22], a way of sampling uniformly on all possible white-box representations of the same black-box function could equally do the job.

The minimal canonical form gives a way to extend this idea trivially to general PEPS. If the presentation (white-box) of the PEPS obtained in the training process is its minimal canonical form, sampling uniformly on all possible white-boxes amounts to sampling with the Haar measure on the unitary group, which can easily be done (as opposed to sampling on the whole general linear group). It is an interesting open question to see how this idea works in practice for PEPS. For MPS it is shown in [PHM+22] that privacy improvements in practice are indeed dramatic.

As alluded to in Section 3.3.4 another natural direction of inquiry is to find physically relevant models where there is topological order which is only revealed on manifolds other than a torus, and see how this relates to the minimal canonical form. Finally, it would be interesting to connect to recent approaches that apply techniques from algebraic geometry and *algebraic complexity theory* [BCS13] to tensor network theory, for instance [CLVW20; CGFW21]. There are also various

⁶Though the inherent continuous nature of the variables makes the problem slightly different in this case.

concrete follow-up questions concerning properties of the minimal canonical form and generalizations.

- (i) **Non-uniform PEPS.** In this work we have mainly restricted to *uniform* PEPS, where we consider contractions of copies of a single identical tensor. We also saw one example with non-uniform tensors, for MPS in Section 3.2.3. In that case, we were able to recover the usual theory of canonical forms for MPS with open boundary conditions. Clearly, an interesting direction for future research is to investigate generalizations of the minimal canonical form to non-uniform PEPS. In this case we consider a fixed graph $\Gamma = (V, E)$, with a collection of tensors $(T_v)_{v \in V}$ at each vertex and where we contract along the edges E . We now have a group $GL(D_e)$ acting on each edge e in the graph, so the full gauge group G is the product over all edges $e \in E$ of these groups. This setup is very similar to the one described in Section 3.2.3. We would like to formulate an appropriate minimization problem over a group orbit. There are two obvious ways to approach this. The first option is to minimize

$$\sum_{v \in V} \|g \cdot T_v\|^2$$

and define a minimal canonical form $((T_v)_{v \in V})_{\min}$ as satisfying

$$((T_v)_{v \in V})_{\min} = \operatorname{argmin} \left\{ \sum_{v \in V} \|S_v\|^2 : (S_v)_{v \in V} \in \overline{G \cdot (T_v)_{v \in V}} \right\}$$

In the case where all tensors are equal, this should reduce to the minimal canonical form for uniform PEPS. A second option (which is similar to the MPS construction in Section 3.2.3) would be to consider for each edge e the tensor network state $|T_e\rangle$ where we have contracted all edges except e . We have a group action of $GL(D_e)$ on this state, and we may minimize over its orbit. We will report on these generalizations in future work.

- (ii) **Algorithms for deciding gauge equivalence.** While we have addressed the issue of computing a minimal canonical form for a given tensor, we have not extensively discussed algorithms for deciding whether two tensors S and T are gauge equivalent. One approach is given by Result 4: one may simply check that $|S_\pi\rangle = |T_\pi\rangle$ for all $\pi \in S_n^m$ with $n \leq n_{\max} = \exp(\mathcal{O}(mD^2 \log(mD)))$ (or in the case of MPS, for $n \leq D^2$). However, an alternative strategy is as follows. By Theorem 3.3.7, it suffices to first compute minimal canonical forms S_{\min} and T_{\min} (for which we have already provided algorithms) and then determine whether these are related by *unitary* gauge transformations (which is rather nontrivial). For $m = 1$, this strategy has been implemented in [AGL+18], while for $m \geq 2$ we defer to future work.
- (iii) **Computational complexity.** It would be interesting to relate the computation of minimal canonical forms and of checking gauge equivalence to other orbit problems that have recently been studied intensely in the theoretical computer science literature, in order to get a better understanding of the computational complexity of the problem (see [BFG+19] and references therein).

Part II.

Interior-point methods for scaling

4. An introduction to interior-point methods

This chapter provides a gentle introduction to the theory of *interior-point methods* for convex programming in Euclidean space. We focus on intuition and omit most proofs here, both for readability and to avoid redundancy, as we extend this theory (with detailed proofs) to geodesically convex objectives on Riemannian manifolds in Chapter 7.

The development of interior-point methods is one of the greatest successes in convex optimization, and by now has a long history dating back to the works of Frisch [Fri55], Karmarkar [Kar84a; Kar84b], Gill et al. [GMS+86], Renegar [Ren88] and many others. It led to one of the first polynomial-time algorithms for linear programming (in contrast with the simplex algorithm due to Dantzig [Dan63]), the other being the ellipsoid method due to Khachiyan [Kha80]. In the seminal work of Nesterov and Nemirovskii [NN94], it was shown that the key property to the analysis of interior-point methods is the notion of *self-concordance*. Essentially every convex programming problem is in principle amenable to interior-point methods, which follows from constructions of self-concordant barriers for arbitrary (bounded) convex domains, cf. [NN94; Hil14; Fox15; BE19; Che23]. Furthermore, interior-point methods are eminently practical, and currently give the best algorithms for linear programming [LS20; Bra19].

This chapter is structured as follows. In Section 4.1 we explain the general idea of interior-point methods. Section 4.2 introduces the most important part of the formalism, notably that of *self-concordant barriers* for a convex domain. Lastly, Section 4.3 discusses the general algorithm for solving optimization problems on convex domains with a self-concordant barrier. We follow the exposition in [Ren01]; other useful sources on interior-point methods are [NN94; Nes18]. We also refer to [BV04] as a general source on convex optimization techniques.

4.1. The idea

We recall the standard interior-point formalism for solving a convex program

$$\begin{aligned} & \text{minimize} && \langle c, p \rangle \\ & \text{subject to} && p \in D, \end{aligned} \tag{4.1.1}$$

where $c \in E$ and $D \subseteq E$ is a closed convex set in some Euclidean space E , and $\langle \cdot, \cdot \rangle$ is the inner product on E . Note that this captures all convex programming problems, by the *epigraph* construction: if $f: D \rightarrow \mathbb{R}$ is some convex function, then we can rewrite $\min_{p \in D} f(p) = \min_{(p,t) \in E_f} t$, where

$$E_f = \{(p, t) \in D \times \mathbb{R} : f(p) \leq t\}$$

This chapter is adapted from [BLNW20; HNW23].

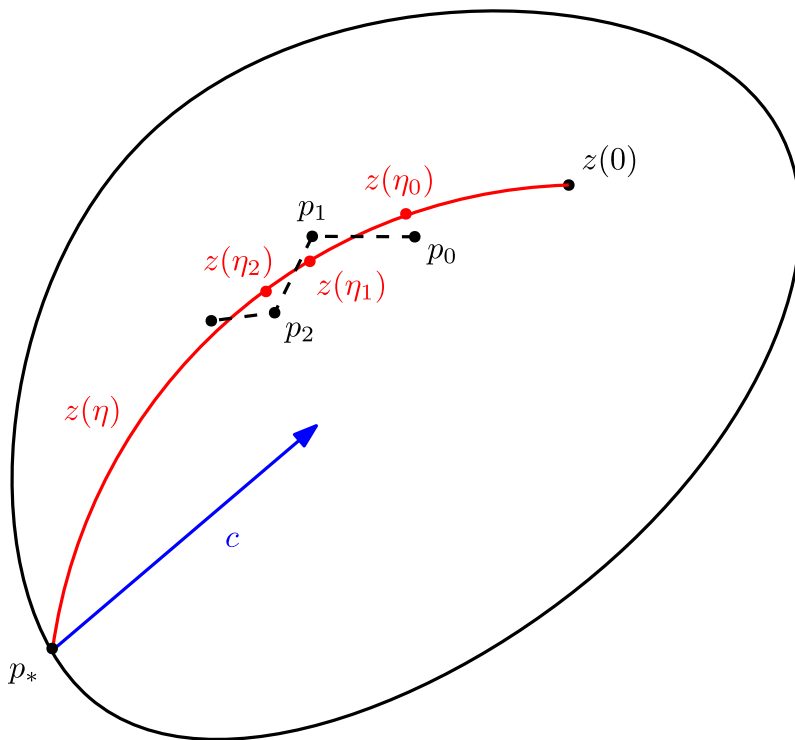


Figure 4.1.: A schematic representation of following the central path.

is the *epigraph* of f , which is a convex domain by convexity of f . For convenience we also assume that D is bounded and has non-empty interior.

The idea of the interior-point methods is then as follows. Since D is bounded, the objective in Eq. (4.1.1) is bounded, and the optimum is attained at some $p_* \in \partial D$ in the boundary. However, in general it may be hard to search directly over ∂D . One notable case in which this is done is Dantzig's simplex method for linear programming [Dan63], but this does not in general yield efficient algorithms [KM72]. Instead, an interior-point method iteratively produces points in the *interior* of the domain of optimization.

More formally, the setup is as follows. Assume one has a *barrier functional* $\Psi: \text{int}(D) \rightarrow \mathbb{R}$, which is a strictly convex function such that $\Psi(p_i) \rightarrow \infty$ as $\text{int}(D) \ni p_i \rightarrow p \in \partial D$. Consider the *central path*, consisting of the minimizers $z(\eta) \in \text{int}(D)$ of the self-concordant functionals

$$\Psi_\eta(p) := \eta \langle c, p \rangle + \Psi(p)$$

for every $\eta \in \mathbb{R}_{>0}$. These minimizers exist, since D is bounded and Ψ blows up at the boundary of D ; moreover, the $z(\eta)$ are unique, since Ψ is strictly convex. Morally speaking, one should also expect that as $\eta \rightarrow \infty$, $z(\eta)$ converges to p_* : after all, minimizers of Ψ_η are also minimizers of $\langle c, p \rangle + \frac{1}{\eta} \Psi(p)$, and for large η the second term is less relevant. Therefore, to solve Eq. (4.1.1), it should suffice to approximately find $z(\eta_i)$ for an increasing sequence of η_i , which is in some sense an *easier* problem: Ψ_η is strictly convex (because Ψ is), and one has a good initial guess for $z(\eta_i)$, namely $z(\eta_{i-1})$, so local optimization methods are likely to be useful.

4.2. Self-concordant barriers

Before we make the above precise, we fix some notation and language. We denote by val the optimal value of Eq. (4.1.1), and we say $p_\delta \in D$ is a δ -minimizer of Eq. (4.1.1) if

$$\langle c, p_\delta \rangle \leq \text{val} + \delta.$$

We say a function $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ is twice continuously differentiable if its gradient and Hessian, which we shall always denote by

$$g(p) = \text{grad } \Psi(p), \quad H(p) = \text{Hess } \Psi(p),$$

are well-defined at any point $p \in \text{int}(D)$, and $H(p)$ depends continuously on p . Recall that a function ψ is convex if for any $p, p' \in \text{int}(D)$ and $t \in (0, 1)$, one has

$$\psi(tp + (1-t)p') \leq t\psi(p) + (1-t)\psi(p'),$$

and strictly convex if the inequality is strict for $t \in (0, 1)$. An equivalent criterion for convexity is that the Hessian $H(p)$ is positive semidefinite for all $p \in \text{int}(D)$, and a sufficient criterion for strict convexity is that $H(p)$ is positive definite for all $p \in \text{int}(D)$. In this case, the Hessian $H(p)$ defines a *local norm* on E for any $p \in D$: for $v \in E$, we write

$$\|v\|_{\Psi, p} = \sqrt{\langle v, H(p)v \rangle}.$$

We also write $B_p^\circ(p, 1) = \{p' \in E : \|p' - p\|_{\Psi, p} < 1\}$ for the *Dikin ellipsoid*, which is the *open ball* with radius 1 centered at some point $p \in D$, measured in the local norm $\|\cdot\|_{\Psi, p}$ at the same point.

To ensure that one can follow the central path and to justify the claim that $z(\eta)$ should converge to a minimizer, one needs to impose additional conditions on Ψ . The *following* notion is *central* in the theory of interior-point methods.

Definition 4.2.1. Let $D \subseteq E$ be closed convex set with non-empty interior. A (*strongly non-degenerate*) *self-concordant barrier functional* for D is a strictly convex and twice continuously differentiable function $\Psi: \text{int}(D) \rightarrow \mathbb{R}$, satisfying the following additional properties:

- (i) For any $p \in \text{int}(D)$, the open ball $B_p^\circ(p, 1)$ is contained in $\text{int}(D)$. Moreover, for any $p' \in B_p^\circ(p, 1)$, we have

$$1 - \|p' - p\|_{\Psi, p} \leq \frac{\|v\|_{\Psi, p'}}{\|v\|_{\Psi, p}} \leq \frac{1}{1 - \|p' - p\|_{\Psi, p}} \quad \text{for all } v \in E \setminus \{0\}.$$

- (ii) The *parameter* ν of the barrier, defined by

$$\nu := \sup \{\|H(p)^{-1}g(p)\|_{\Psi, p}^2 : p \in \text{int}(D)\},$$

is finite.

Roughly speaking, property (i) guarantees that the Hessian $H(p)$ does not change too quickly, when one takes small steps as measured in the norm induced by $H(p)$. This estimate on its own is usually referred to as *self-concordance* of a function, and provides rigorous guarantees on the performance of Newton's method (as we shall explain later). One of the key reasons the notion of self-concordance was introduced is that it is invariant under affine reparameterization: traditional analyses of Newton's method tend to assume that the Hessian of the objective is Lipschitz, but such an estimate is not scale-invariant.

Property (ii) says that the gradient is uniformly bounded with respect to the local norm, i.e., the parameter is naturally a type of "Lipschitz constant". It primarily plays a role in the speed at which one can follow the central path, as we discuss below in Section 4.3.

One can show that a self-concordant barrier Ψ automatically blows up at the boundary of its domain: if $p_i \in \text{int}(D)$ converges to $\bar{p} \in \partial D$, then $\Psi(p_i) \rightarrow \infty$ and $\|g(p_i)\|_2 \rightarrow \infty$ for $i \rightarrow \infty$; see [Ren01, Thm 2.2.9]. This justifies calling it a barrier for the domain.

An important property of self-concordant barriers as defined above is that they satisfy a certain *barrier calculus*. This means that given barriers for a domain, it is easy to construct new barriers. For instance, affine reparameterization or restrictions, products of domains, and intersections of domains all admit self-concordant barriers whenever each of the components admits one. The parameter of the barrier also behaves well under these operations: for both products and intersections, the parameter of the resulting barrier is the sum of the parameters of its constituents.

We note here that (i) is usually difficult to verify directly. The original definition of self-concordance due to Nesterov and Nemirovskii [NN94, Def. 2.1.1] is as follows: a C^3 -smooth convex function $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ is self-concordant if for all $p \in \text{int}(D)$ and $u \in E$, the function $\phi(t) = \Psi(p + tu)$ satisfies

$$|\phi'''(t)| \leq 2(\phi''(t))^{3/2},$$

or equivalently,

$$|D^3\Psi(p)[u, u, u]| \leq 2D^2\Psi(p)[u, u]^{3/2}. \quad (4.2.1)$$

That this definition is equivalent (for C^3 -smooth functions) is shown in [Ren01, Sec. 2.5]. However, it is usually far more tractable to verify this estimate, and in Chapter 7 we shall actually use this perspective. We provide some examples of self-concordant barriers here:

Example 4.2.2 (Linear programming). An instructive example is the barrier $\Psi(p) = -\log p$ for the half-line $\mathbb{R}_{\geq 0} \subseteq \mathbb{R}$, which has parameter $\nu = 1$. To see that Ψ is self-concordant, we compute for $p > 0$:

$$\Psi'(p) = -\frac{1}{p}, \quad \Psi''(p) = \frac{1}{p^2}, \quad \Psi'''(p) = -\frac{2}{p^3}.$$

Therefore $|\Psi'''(p)| = 2\Psi''(p)^{3/2}$ holds exactly. Furthermore, the parameter is 1, since $(\Psi'(p))^2/\Psi''(p) = 1$.

Together with the barrier calculus described above, self-concordant barriers for all convex polytopes can be obtained, given a description of the polytope as a finite

intersection of hyperplanes. This implies that interior-point methods can be used to solve linear programming problem: a self-concordant barrier for the set of $x \in \mathbb{R}^n$ described by inequalities $\langle a_j, x \rangle \leq b_j$ for $a_1, \dots, a_k \in \mathbb{R}^n$ and $b_1, \dots, b_k \in \mathbb{R}$ is given by

$$\Psi(x) = \sum_{j=1}^k -\ln(b_j - \langle a_j, x \rangle).$$

Example 4.2.3 (Semidefinite programming). Let $D = \text{SPD}(n, \mathbb{C})$ denote the cone of positive-semidefinite matrices with entries in \mathbb{C} . Then $\Psi: \text{int}(D) = \text{PD}(n, \mathbb{C}) \rightarrow \mathbb{R}$ given by $\Psi(P) = -\log \det(P)$ is a self-concordant barrier with parameter $\nu = n$ [NN94, Prop. 5.4.5]. This barrier can be used for semidefinite programming problems.

Example 4.2.4 (Second-order cone). Let $n \geq 1$ and $D = \{(x, r) \in \mathbb{R}^n \times \mathbb{R} : \|x\|_2 \leq r\}$. Then $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ defined by

$$\Psi(x, r) = -\log(r^2 - \|x\|_2^2)$$

is a self-concordant barrier with parameter $\nu = 2$ [NN94, Prop. 5.4.3]; in particular independent, the parameter is independent of n . This barrier is particularly useful for enforcing norm-constraints on domains.

In the context of unconstrained geometric programming, see Chapter 5, the following barrier is instrumental:

Example 4.2.5 (Exponential cone). Let $D = \{(y, z) \in \mathbb{R}^2 : e^y \leq z\}$ be the *exponential cone*. Then $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ defined by

$$\Psi(y, z) = -\log z - \log(\log(z) - y)$$

is a self-concordant barrier for D , with parameter $\nu = 2$. Proving this is non-trivial even with Eq. (4.2.1), see [NN94, Prop. 5.3.3].

Example 4.2.6 (Operator logarithm hypograph). A natural *non-commutative* extension of the exponential cone would be the set

$$D' = \{(Y, Z) \in \text{Herm}(n)^2 : e^Y \preceq Z\}$$

where $e^Y \preceq Z$ refers to the Löwner ordering, i.e., $Z - e^Y$ is positive semidefinite. However, the exponential function is not operator-convex [Bha13, Prob. V.5.1], so D' is not a Euclidean convex set. This is also a setback in the context of non-commutative norm minimization problems, as we explain now. For a representation $\pi: \mathfrak{G} \rightarrow \text{GL}(V)$ and $v \in V \setminus \{0\}$ as in Section 2.6, observe that

$$\inf_{g \in \mathfrak{G}} \|g \cdot v\|^2 = \inf_{g \in \mathfrak{G}} \langle v, \pi(g^*g)v \rangle = \inf_{X \in \mathfrak{iLie}(\mathfrak{K})} \langle v, e^{\Pi(X)}v \rangle \quad (4.2.2)$$

since $\pi(g^*g) = \pi(e^X) = e^{\Pi(X)}$ for some $X \in \mathfrak{iLie}(\mathfrak{K})$ (by the polar decomposition, Theorem 2.2.16), where $\Pi = d\pi_1: \text{Lie}(\mathfrak{G}) \rightarrow \text{Lie}(\text{GL}(V))$ is the induced map on Lie algebras. If D' were convex, we could write Eq. (4.2.2) as a convex optimization problem over D' with the linear constraint $Y = \Pi(X)$ for some X .

By contrast, the *hypograph* of the operator logarithm is convex, i.e.,

$$D = \{(Y, Z) \in \text{Herm}(n)^2 : Y \preceq \log(Z), Z \succeq 0\}$$

is a convex set, because the operator logarithm is operator-concave [Bha09, Ex. 4.2.5]. This domain admits the self-concordant barrier

$$\Psi(Y, Z) = -\log \det(Z) - \log \det(Y - \log(Z)),$$

with barrier parameter $2n$. More generally, for operator-monotone functions, one can construct barriers for their epigraphs [FZ20; FS22].

4.3. Following the central path

Under the assumption that Ψ is a self-concordant barrier functional, one can show that one can indeed follow the central path, consisting of the minimizers $z(\eta)$ of $\Psi_\eta = \eta \langle c, \cdot \rangle + \Psi$. One follows the central path along a sequence of $\eta_1 < \eta_2 < \dots$ which grows geometrically, with a rate depending on the parameter $\nu \geq 0$ of the barrier. To find $z(\eta_i)$ starting from $z(\eta_{i-1})$, one uses Newton's method as applied to Ψ_{η_i} , whose behaviour is controlled by using the self-concordance of the barrier functional (Definition 4.2.1(i)). It is also well-known [Ren01, (2.12)] that

$$\langle c, z(\eta) \rangle \leq \text{val} + \frac{\nu}{\eta}, \quad (4.3.1)$$

so following the central path as $\eta \rightarrow \infty$ guarantees convergence of $z(\eta)$ to a minimizer p_* of the objective.

A precise description of the main stage is given in Algorithm 4.1, and a schematic representation is given in Fig. 4.1. One assumes to have a starting parameter η_0 , and a starting point $p_0 \in \text{int}(D)$, which is an approximate minimizer of Ψ_{η_0} , so in particular close to $z(\eta_0)$. Then, for $i \geq 1$, one chooses an appropriate $\eta_i > \eta_{i-1}$ such that a single Newton step for the function Ψ_{η_i} at point p_{i-1} produces a point p_i that is guaranteed to remain close to the central path. Since $\text{grad } \Psi_{\eta_i}(p) = \eta_i c + \text{grad } \Psi(p) = \eta_i c + g(p)$ and $\text{Hess } \Psi_{\eta_i}(p) = \text{Hess } \Psi(p) = H(p)$, the point p_i obtained by taking a single Newton step is given by

$$p_i = p_{i-1} - (\text{Hess } \Psi_{\eta_i}(p_{i-1}))^{-1} \text{grad } \Psi_{\eta_i}(p_{i-1}) = p_{i-1} - H(p_{i-1})^{-1}(\eta_i c + g(p_{i-1})).$$

If we write

$$\alpha_i(p) := \|H(p)^{-1}(\eta_i c + g(p))\|_{\Psi, p},$$

then the length of the Newton step, measured in the local norm at p_{i-1} , is $\alpha_i(p_{i-1})$. Furthermore, one can show that $\alpha_i(p)$ is directly related to the distance of p to the minimizer $z(\eta_i)$ of Ψ_{η_i} (cf. [Ren01, Thm. 2.2.5]). Therefore, by choosing the η_i such that $\alpha_i(p_i)$ stays small, we guarantee that the iterates p_i remain close to the central path. This is achieved by first estimating $\alpha_i(p_{i-1})$ in terms of $\alpha_{i-1}(p_{i-1})$, the ratio η_i/η_{i-1} , and the parameter ν of the barrier, and then bounding $\alpha_i(p_i)$ in terms of $\alpha_i(p_{i-1})$ using self-concordance [Ren01, (2.15)–(2.16)]. Provided $\eta_i \rightarrow \infty$ as $i \rightarrow \infty$, Eq. (4.3.1) suggests that the p_i converge to a minimizer of the objective. A suitable choice of the η_i , along with a quantitative guarantee on the precision achieved by any particular p_i is given by the following theorem.

Algorithm 4.1: MainStage

Input: starting point $p_0 \in D$, starting parameter $\eta_0 > 0$, objective $c \in E$, iteration count $T \geq 0$, parameter $\nu \geq 1$ and oracle access to gradient $g(p)$ and Hessian $H(p)$ of barrier $\Psi: \text{int}(D) \rightarrow \mathbb{R}$

```

1 for  $i = 1, \dots, T$  do
2    $\eta_i \leftarrow \left(1 + \frac{1}{8\sqrt{\nu}}\right) \eta_{i-1}$ ;
3    $p_i \leftarrow p_{i-1} - H(p_{i-1})^{-1} (\eta_i c + g(p_{i-1}))$ ;            $\triangleright$  Newton step for  $\Psi_{\eta_i}$  at  $p_{i-1}$ 
4 end for
5 return  $p_T$ ;

```

Theorem 4.3.1 (Main stage, [Ren01, p. 46–47, (2.14), and (2.17)]). *Let $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ be a strongly non-degenerate self-concordant barrier functional for D with parameter $\nu \geq 1$. Let $\eta_0 > 0$ be given, and suppose $p_0 \in \text{int}(D)$ satisfies*

$$\alpha_0(p_0) = \|H(p_0)^{-1}(\eta_0 c + g(p_0))\|_{\Psi, p_0} \leq \frac{1}{9}. \quad (4.3.2)$$

Then the iterations of Algorithm 4.1 are well-defined and we have, for all $i \in [T]$, that $\alpha_i(p_i) \leq \frac{1}{9}$, $\|p_i - z(\eta_i)\|_{(z(\eta_i))} \leq \frac{1}{5}$, and $\langle c, p_i \rangle \leq \text{val} + \frac{6}{5\eta_i} \nu$.

In particular, for $T \geq 10\sqrt{\nu} \log\left(\frac{6}{5} \frac{\nu}{\eta_0 \delta}\right)$, Algorithm 4.1 returns a point $p_T \in \text{int}(D)$ satisfying

$$\langle c, p_T \rangle \leq \text{val} + \delta.$$

One issue with the above strategy is that one has to know a good starting point: after all, one cannot expect to follow the central path without starting close to it! Finding such a starting point is the purpose of the *preliminary stage*. A precise definition of a good starting point (and starting parameter) is satisfying the hypotheses of Theorem 4.3.1. An algorithm that achieves this is presented in Algorithm 4.2. One starts from an *arbitrary* point $p'_0 \in \text{int}(D)$ and follows the central path associated with the objective $-g(p'_0)$ and the same self-concordant barrier. This objective is chosen because p'_0 is the minimizer of $-\mu \langle g(p'_0), p \rangle + \Psi(p)$ when $\mu = 1$, i.e., p'_0 is exactly on the central path at time 1. Now one *decreases* the parameter μ , rather than increasing it, until one obtains an approximate minimizer of $\Psi = \Psi_0$. Finally, one chooses an appropriate $\eta_0 > 0$ and performs a single Newton step for Ψ_{η_0} that is guaranteed to yield a point p_0 satisfying Eq. (4.3.2). Only this last step depends on the objective c of the convex program Eq. (4.1.1). The following definition and theorem bound the number of iterations of Algorithm 4.2 and give a lower bound on η_0 .

Definition 4.3.2 (Symmetry). Let $D \subseteq E$ be a compact convex subset, and let $p \in \text{int}(D)$. The *symmetry* of D with respect to p is defined by

$$\text{sym}(p) = \max \{a \geq 0 : p + a(p - D) \subseteq D\}.$$

If L is an affine line through p , then $L \cap D$ consists of two chords from p to the boundary of D ; the symmetry parameter $\text{sym}(p)$ is the smallest possible ratio of the lengths of the smallest and longest chord. Therefore, the symmetry is always at

Algorithm 4.2: PreliminaryStage

Input: starting point $p'_0 \in D$, objective $c \in E$, parameter $\nu \geq 1$ and oracle access to gradient $g(p)$ and Hessian $H(p)$ of barrier $\Psi: \text{int}(D) \rightarrow \mathbb{R}$

- 1 $\mu_0 \leftarrow 1$;
- 2 $g_0 \leftarrow g(p'_0)$;
- 3 $i \leftarrow 0$;
- 4 **while** $\|H(p'_i)^{-1}g(p'_i)\|_{\Psi, p'_i} > \frac{1}{6}$ **do**
- 5 $i \leftarrow i + 1$;
- 6 $\mu_i \leftarrow \left(1 - \frac{1}{8\sqrt{\nu}}\right) \mu_{i-1}$;
- 7 $p'_i \leftarrow p'_{i-1} - H(p'_{i-1})^{-1}(-\mu_i g_0 + g(p'_{i-1}))$; \triangleright Newton step for $-\mu_i g_0 + \Psi$ at p'_{i-1}
- 8 **end while**
- 9 $\eta_0 \leftarrow (12\|H(p'_i)^{-1}c\|_{\Psi, p'_i})^{-1}$;
- 10 $p_0 \leftarrow p'_i - H(p'_i)^{-1}(\eta_0 c + g(p'_i))$; \triangleright Newton step for Ψ_{η_0} at p'_i
- 11 **return** (p_0, η_0) ;

most 1, and from this description, it is also clear that one can bound the symmetry by providing a ball centered at p contained in the interior of D , and another ball centered at p containing all of D ; see Lemma 5.4.3.

Theorem 4.3.3 (Preliminary stage, [Ren01, (2.19)]). *Let $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ be a strongly non-degenerate self-concordant barrier functional for D with parameter $\nu \geq 1$, let $p'_0 \in \text{int}(D)$ be a starting point, and let $c \in E$ be the objective. Then Algorithm 4.2 with this choice of starting point p'_0 outputs a vector $p_0 \in \text{int}(D)$ and $\eta_0 > 0$ satisfying Eq. (4.3.2), i.e.,*

$$\|H(p_0)^{-1}(\eta_0 c + g(p_0))\|_{\Psi, p_0} \leq \frac{1}{9},$$

as soon as $\mu_i^{-1} \geq 18\nu(1 + 1/\text{sym}(p'_0))$, i.e., after at most

$$\frac{\log(18\nu(1 + \frac{1}{\text{sym}(p'_0)}))}{-\log(1 - \frac{1}{8\sqrt{\nu}})} \leq 8\sqrt{\nu} \log\left(\frac{36\nu}{\text{sym}(p'_0)}\right)$$

iterations. Moreover, we have the lower bound $\eta_0 \geq \frac{1}{12(V - \text{val})}$, where $V = \max_{p \in D} \langle c, p \rangle$.

Together, Theorems 4.3.1 and 4.3.3 can be summarized as follows:

Theorem 4.3.4 (Theorem 2.4.1 in [Ren01]). *Let $D \subseteq E$ be a compact convex subset with non-empty interior. Assume $\Psi: \text{int}(D) \rightarrow \mathbb{R}$ is a strongly non-degenerate self-concordant barrier functional for D with parameter $\nu \geq 1$. Furthermore, for $c \in E$, define $\text{val} = \min_{p \in D} \langle c, p \rangle$ and $V = \max_{p \in D} \langle c, p \rangle$. Finally, let $0 < \delta < V - \text{val}$ be the desired precision and let $p'_0 \in \text{int}(D)$ be a starting point for the preliminary stage. Then, Algorithm 4.2 outputs a point $p_0 \in \text{int}(D)$ and a parameter $\eta_0 \geq \frac{1}{12(V - \text{val})}$ satisfying the hypotheses of Theorem 4.3.1. Algorithm 4.1 with inputs p_0, η_0 and $T \geq 10\sqrt{\nu} \log(\frac{6}{5} \frac{\nu}{\eta_0 \delta})$ outputs a point p_T satisfying*

$$\langle c, p_T \rangle - \text{val} \leq \delta.$$

The total number of iterations is upper bounded by

$$18\sqrt{\nu} \log \left(\frac{36\nu}{\text{sym}(p'_0)} \frac{V - \text{val}}{\delta} \right)$$

and each iteration involves computing the gradient and Hessian of the self-concordant barrier Ψ and basic matrix arithmetic.

5. Interior-point methods for commutative scaling problems

In this chapter we show that the interior-point framework from Chapter 4 is useful in the context of the norm-minimization and scaling problems as defined in Section 2.6, when the acting group is *commutative*. In this case, the underlying objective is convex on Euclidean space: more specifically, it reduces to an unconstrained geometric programming problem. It serves as a useful example, showing what kind of challenges arise in applying interior-point methods to scaling problems, and sets a baseline for results one may hope to achieve in the general non-commutative setting.

The chapter is organized as follows. In Section 5.1 give a detailed introduction to (unconstrained) geometric programming and the relation to commutative scaling. In Section 5.2 we provide a brief summary of the results: we define the *geometric condition measures* and the *facet gap*, and state the IPM iteration complexity results. In Section 5.3 we discuss the condition numbers defined in Section 5.2 in more detail and show how they imply diameter bounds on (approximate) minimizers of the GP. In Section 5.4 we explain how to use these diameter bounds together with the general framework of interior-point methods to prove Theorems 5.2.2 and 5.2.5 and their corollaries. Lastly, in Section 5.5, we give a priori bounds on the condition numbers in terms of the encoding length of the input and we also provide better condition number bounds when the geometric program is totally unimodular. These bounds imply that the iteration complexity of the interior-point methods is *polynomial* in the input size.

5.1. Introduction

Geometric programming is an optimization paradigm that generalizes linear programming and has a wide range of applications [DPZ67; BKH07]. We shall concern ourselves only with *unconstrained geometric programs*. These are optimization problems of the form

$$\begin{aligned} & \text{minimize} && f(z) \\ & \text{subject to} && z \in \mathbb{R}_{>0}^n, \end{aligned} \tag{5.1.1}$$

where $f(z)$ is a *posynomial* in positive real variables z_1, \dots, z_n . That is,

$$f(z) = \sum_{i=1}^k q_i z^{\omega_i} = \sum_{i=1}^k q_i \prod_{j=1}^n z_j^{\omega_{i,j}}, \tag{5.1.2}$$

This chapter is adapted from [BLNW20].

where the coefficients q_i are positive and the exponents $\omega_{i,j}$ are real numbers. In a general geometric program (GP), one adds posynomial inequality and monomial equality constraints. Although posynomials are non-convex in general, they are convex in $x \in \mathbb{R}^n$ after the change of variables $z_i = e^{x_i}$. This means they are the simplest family of *geodesically convex* programming problems.

Deciding whether an unconstrained geometric program is bounded from below captures membership of a vector in a convex hull, as we shall see shortly. In this sense, unconstrained GP can be viewed as a generalization of linear programming.¹ Therefore simple algorithms like gradient descent or Newton's method are unlikely to yield efficient algorithms. However, it is also well known that unconstrained geometric programs can be solved in polynomial time via the ellipsoid method, see [NR99].

It seems common wisdom that GP can be solved in polynomial time via interior-point methods, however we were unable to find a rigorous justification of this claim in the literature, see for instance [NN94; KK96; NR99; KXY97; AY98; BV04; NT05; KT18]. To remedy this, we provide a systematic treatment of unconstrained GP, along with detailed complexity bounds for two interior-point algorithms for unconstrained GP in terms of natural geometric condition numbers. Our first algorithm applies to instances that (roughly speaking) have a well-conditioned Newton polytope, while our second algorithm has no such assumption but instead relies on a novel condition number for the GP.

We also provide effective bounds on the condition numbers for rational inputs. Our results improve over the optimization algorithms of [BFG+19], which apply to general non-commutative norm minimization and scaling problems. Under additional combinatorial assumptions, satisfied for instance in the case of matrix scaling and balancing, we match the iteration complexity of the interior-point methods provided in [CMTV17].

5.1.1. The computational problems

We define the computational problems associated with Eq. (5.1.1) in a more formal way, specializing the norm minimization and scaling problems from Section 2.6. We write $q = (q_1, \dots, q_k) \in \mathbb{R}_{>0}^k$ for the vector of *coefficients* and $\Omega = \{\omega_1, \dots, \omega_k\} \subseteq \mathbb{R}^n$ for the set of *exponents* of the posynomial Eq. (5.1.2).

After the reparameterization $z_i = e^{x_i}$, the objective takes the form

$$\sum_{j=1}^k q_j z^{\omega_j} = \sum_{j=1}^k q_j e^{\langle \omega_j, x \rangle}. \quad (5.1.3)$$

This is in fact even *log-convex* in x . It is convenient to give its logarithm a name, and to allow an overall shift of the ω_j 's by a fixed vector $\theta \in \mathbb{R}^n$: we define

$$F_\theta(x) := \log \left(\sum_{j=1}^k q_j e^{\langle \omega_j - \theta, x \rangle} \right) = \log \left(\sum_{j=1}^k q_j e^{\langle \omega_j, x \rangle} \right) - \langle \theta, x \rangle, \quad F_\theta^* := \inf_{x \in \mathbb{R}^n} F_\theta(x). \quad (5.1.4)$$

¹General geometric programming is even a direct generalization: when the objective and constraints are all *monomials*, the change of variables turns the GP into a general linear program.

When the ω_j are integer vectors, expressions like the above (for $\theta = 0$) arise as Kempf–Ness functions for commutative groups; in fact, by virtue of the weight decomposition (Theorem 2.2.13), this captures all *commutative* norm minimization problems. More precisely, if $G = (\mathbb{C}^\times)^n$ acts on $V = \mathbb{C}^\Omega$ via

$$(w_1, \dots, w_n) \cdot e_\omega = w_1^{\omega_1} \cdots w_n^{\omega_n} e_\omega,$$

where $\{e_\omega\}$ is the canonical orthonormal basis for V , then for $v \in V \setminus \{0\}$, we have

$$\log \|w \cdot v\| = \frac{1}{2} \log \left(\sum_{\omega \in \Omega} |v_\omega|^2 |w^\omega|^2 \right).$$

Reparameterizing $|w_i|^2 = e^{x_i}$ and $q_\omega = |v_\omega|^2$ yields the unconstrained GP of the form Eq. (5.1.4) (up to the prefactor 2).

We note that the notation F_θ should not be confused with the notation “ F_v ” for the Kempf–Ness function from Section 1.2 and Chapter 2. The role of $v \in V$ is now played by the vector q , which we view as fixed. Instead, we are primarily interested in the behavior of algorithms as $\theta \in \mathbb{R}^n$ changes; in particular for the question “is θ in the Newton polytope?” (with the usual scaling problem corresponding to $\theta = 0$).

The problem of unconstrained geometric programming is to approximate the infimum to arbitrary precision:

Problem 5.1.1 (Unconstrained GP with shift). *Given as input exponents $\omega_1, \dots, \omega_k \in \mathbb{R}^n$, a shift $\theta \in \mathbb{R}^n$, $q \in \mathbb{R}_{>0}^k$, and a precision $\delta > 0$, find $x_\delta \in \mathbb{R}^n$ such that $F_\theta(x_\delta) \leq F_\theta^* + \delta$.*

Clearly, any solution to this problem provides a $(1 + 2\delta)$ -multiplicative and a $(2\|q\|_1\delta)$ -additive approximation to the value of the original geometric program Eq. (5.1.1).

Problem 5.1.1 depends crucially on the *Newton polytope* of f , which is defined as the convex hull of the set of exponents Ω . This is exactly the moment polytope for the commutative setting, see Section 2.5. A slight refinement of the Kempf–Ness theorem (Theorems 2.4.4 and 2.5.5) in this setting is then:

Proposition 5.1.2. *The function F_θ defined in Eq. (5.1.4) satisfies:*

- (i) $F_\theta^* > -\infty$ if and only if $\theta \in \text{conv}(\Omega)$, and in this case $F_\theta^* \geq \log \min_{i \in [k]} q_i$,
- (ii) $F_\theta^* = F_\theta(x)$ for some $x \in \mathbb{R}^n$ if and only if $\theta \in \text{relint}(\text{conv} \Omega)$, where $\text{relint}(\cdot)$ denotes the relative interior.

Proof. Property (i) follows from the observation that F_θ is unbounded from below if and only if there exists some $x \in \mathbb{R}^n$ such that $\langle \omega_i - \theta, x \rangle < 0$ for all $i \in [k]$. This is in turn equivalent to $\theta \notin \text{conv} \Omega$ by Farkas’ lemma [BV04, Sec. 5.8]. Now in case $\theta \in \text{conv} \Omega$, for every $x \in \mathbb{R}^n$, there exists $j \in [k]$ such that $\langle \omega_j - \theta, x \rangle \geq 0$, and it follows that

$$F_\theta(x) \geq \log(q_j e^{\langle \omega_j - \theta, x \rangle}) \geq \log \min_{i \in [k]} q_i. \quad (5.1.5)$$

One direction of property (ii) follows from the observation that $\text{grad} F(x)$ is always in $\text{relint}(\text{conv} \Omega)$; therefore, if $F_\theta^* = F_\theta(x)$ for some x , we have $\text{grad} F_\theta(x) =$

$\text{grad } F(x) - \theta = 0$ and θ is in $\text{reint}(\text{conv } \Omega)$. The other direction follows from diameter bounds proven later (Proposition 5.3.1): if θ is in the relative interior of $\text{conv } \Omega$, then one can show that every $x \in \mathbb{R}^n$ with $\|x\|_2$ sufficiently large satisfies $F_\theta(x) > F_\theta(0)$. \square

Thus, deciding whether F_θ^* is finite or not can be done by testing membership in the Newton polytope (a linear programming problem). By convexity, Problem 5.1.1 is directly related to the problem of minimizing (the norm of) the gradient $\text{grad } F_\theta(x)$, which is given by

$$\text{grad } F_\theta(x) = \frac{\sum_{j=1}^k q_j e^{\langle \omega_j, x \rangle} \omega_j}{\sum_{j=1}^k q_j e^{\langle \omega_j, x \rangle}} - \theta.$$

We refer to this as the associated *scaling* problem, specializing the definition from Section 2.6.

Problem 5.1.3 (Scaling problem with shift). *Given as input exponents $\omega_1, \dots, \omega_k \in \mathbb{R}^n$, a shift $\theta \in \mathbb{R}^n$, $q \in \mathbb{R}_{>0}^k$, and a precision $\varepsilon > 0$, find $x_\varepsilon \in \mathbb{R}^n$ such that $\|\text{grad } F_\theta(x_\varepsilon)\|_2 = \|\text{grad } F(x_\varepsilon) - \theta\|_2 \leq \varepsilon$.*

The shift θ is useful in various contexts, such as matrix scaling (as discussed in Section 1.1): observe that rather than using the potential function f from Eq. (1.1.1), one could have also used the function

$$F(x_1, \dots, x_n, y_1, \dots, y_n) = \log \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right) - \langle (x, y), (r, c) \rangle.$$

which is essentially an unconstrained geometric program with q 's given by the A_{ij} , ω 's given by $(e_i, e_j) \in \mathbb{R}^n \times \mathbb{R}^n$, and the shift given by the target marginals (r, c) .

5.1.2. Entropy maximization

Minimizing F_θ also has a useful *dual* formulation, which is given by an entropy maximization problem [SV14]. More precisely,

$$F_\theta^* = \inf_{x \in \mathbb{R}^n} F_\theta(x) = \sup \left\{ -D(p \| q) : \sum_{j=1}^k p_j \omega_j = \theta, \sum_{j=1}^k p_j = 1, p \geq 0 \right\}, \quad (5.1.6)$$

where $D(p \| q) = \sum_{j=1}^k p_j \log \frac{p_j}{q_j}$ denotes the *Kullback–Leibler (KL) divergence* between a probability distribution p and the distribution $q = (q_1, \dots, q_k)$ (which need not be normalized). Thus, the dual program Eq. (5.1.6) is feasible (i.e., has non-empty domain) when θ is in the convex hull of the ω_i , which is often referred to as the *Newton polytope* of the unconstrained geometric program. Furthermore, the optimal solution is a probability distribution on $\Omega = \{\omega_j\}$ with mean θ that minimizes the KL divergence to the initial distribution q . When $q = (1, \dots, 1)$ is the all-ones vector, $-D(p \| q) = \sum_{i=1}^k p_i \log \frac{1}{p_i}$ is the Shannon entropy of p . In this case, Eq. (5.1.6) amounts to the discrete *entropy maximization* problem which naturally arises in machine learning and statistics, motivated by the maximum entropy principle [Jay57b; Jay57a]. From this perspective, it is also easy to see the connection between matrix scaling and to entropy-regularized optimal transport [Cut13].

5.1.3. Diameter bounds

To solve the entropy maximization problem Eq. (5.1.6), [SV14; SV19] proposed ellipsoid methods for the equivalent geometric program Eq. (5.1.4) that are tractable even when k is large. They focused on the case that Ω consists of integer vectors (which is already of substantial interest) and gave a priori diameter bounds as required for the ellipsoid method. In [SV14], it was shown that if θ is at a distance $\eta > 0$ from the boundary of the Newton polytope then there is a minimizer x^* of norm $\|x^*\|_2 \leq \frac{\log k}{\eta}$. In [SV19], a diameter bound was obtained in terms of the *unary facet complexity* of the Newton polytope: if $\text{conv}(\Omega)$ can be described by linear inequalities with integer coefficients in $\{-M, \dots, M\}$, then for *any* $\theta \in \text{conv}(\Omega)$ there is a δ -approximate minimizer x_δ of the function F_θ with $\|x_\delta\|_2 \leq R$, where $R = \text{poly}(n, M, \log \frac{1}{\delta})$. This bound is particularly useful if θ is very close to (or on) the boundary of $\text{conv} \Omega$. We generalize both of these bounds to the case where $\Omega \subseteq \mathbb{R}^n$ is not necessarily integral.

5.1.4. Complexity

We comment briefly on the notion of complexity used in this chapter. The stated complexity bounds on the given interior-point methods are given in the number of *iterations*. Each iteration consists however consists of arithmetic operations (computing gradients and Hessians, and solving a linear system for taking Newton steps). Therefore we only get *polynomial-time* algorithms in a (infinite-precision) real-number model of computation [BCSS98]. It is currently unknown whether this can be extended to the Turing-machine model of computation: the obstruction is that it is unclear whether the bit-complexity of the numbers appearing in the algorithm remains bounded or not. In the setting of linear programming this is known not to be an issue [Ren88], and in the context of semidefinite programming such a result is only relatively recent [KV16]. One hint that such a result might hold in our setting is that the updates (given by Newton steps) in the usual IPM framework are well-known to be error-robust, see for instance [CMTV17, Sec. 6.3].

5.1.5. Notation and assumptions

Throughout we will always assume that the shift is contained in the Newton polytope, i.e., $\theta \in \text{conv}(\Omega)$. In order to state our results, we define a quantity capturing the condition of $q \in \mathbb{R}_{>0}^k$:

$$\beta = \frac{\|q\|_1}{\min_{i \in [k]} q_i}, \quad (5.1.7)$$

where $\|q\|_1 = \sum_{i=1}^k q_i$. We refer to q as a distribution, in the sense of a unnormalized probability distribution; the normalization is not important for solving the unconstrained GP, and this is reflected in β being invariant under rescaling q . Note that $k \leq \beta \leq \frac{\max_i q_i}{\min_i q_i} k < \infty$. With this notation, observe that if $\theta \in \text{conv} \Omega$, then by Proposition 5.1.2 we have $F_\theta(0) - F_\theta^* \leq \log \|q\|_1 - \log \min_{i \in [k]} q_i = \log(\beta)$.

We denote by $B(\theta, r)$ the closed ball centered at θ with radius r , and $\text{aff}(\Omega)$ denotes the affine hull of Ω (i.e., the smallest affine subspace of \mathbb{R}^n containing Ω).

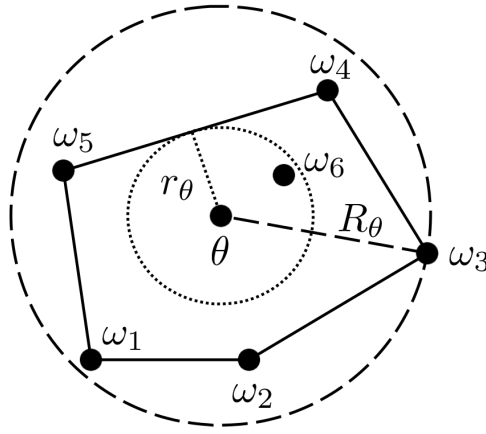


Figure 5.1.: An illustration of r_θ and R_θ .

5.2. Summary of results

5.2.1. Well-conditioned instances

Here we state our results in terms of the natural condition measures r_θ and R_θ defined below; see Fig. 5.1.

Definition 5.2.1 (Geometric condition measures). Given an instance of the unconstrained GP or scaling problem with $\Omega \subseteq \mathbb{R}^n$ and shift $\theta \in \text{conv}(\Omega)$, we define r_θ as the radius of the largest ball about θ contained in the polytope:

$$r_\theta = \max\{r \geq 0 : B(\theta, r) \cap \text{aff}(\Omega) \subseteq \text{conv}(\Omega)\} = d(\theta, \partial \text{conv}(\Omega)).$$

We say that the instance is *well-conditioned* if $r_\theta > 0$, and *ill-conditioned* otherwise. Similarly, we define R_θ as the *radius of the smallest enclosing ball* about θ :

$$R_\theta = \min\{R \geq 0 : \text{conv}(\Omega) \subseteq B(\theta, R)\} = \max_{\omega \in \Omega} \|\omega - \theta\|_2.$$

Thus, an instance is well-conditioned when θ is in the relative interior of the Newton polytope, and ill-conditioned if it is on the boundary. The quantity r_θ is closely related to a condition measure due to [Gof80], which is widely used in the context of testing polyhedral cone feasibility [BC13; DVZ20], see Remark 5.3.2.

Our first result is a bound on the number of iteration steps of a natural interior-point method (IPM), which solves well-conditioned instances of unconstrained GP.

Theorem 5.2.2. *There is an interior-point algorithm (Algorithm 5.2) that, given as input a well-conditioned instance of the unconstrained GP problem with shift (Problem 5.1.1), returns $x_\delta \in \mathbb{R}^n$ such that $F_\theta(x_\delta) \leq F_\theta^* + \delta$ within*

$$O\left(\sqrt{k} \log\left(k \frac{R_\theta}{r_\theta} \frac{1}{\delta} \log(k\beta)\right)\right)$$

iterations. The starting point of the algorithm is determined explicitly by the input, and every iteration is a Newton step.

We emphasize that it is *not* necessary to provide a lower bound on r_θ as input. The algorithm follows the interior-point method framework of [NN94; Ren01], which consists of a *preliminary stage* and a *main stage*. We refer back to Chapter 4 for details, but briefly recall the most important aspects. The preliminary stage uses a starting point that is easily computed in terms of the input data, and outputs a starting point for the main stage within $O(\sqrt{k} \log(k \frac{R_\theta}{r_\theta} \log(k\beta)))$ Newton iterations. The main stage then produces a sequence of points x_0, x_1, \dots such that $F_\theta(x_j) - F_\theta^* \leq C \log(k\beta) \exp(-\frac{j}{\sqrt{k}})$ for some known constants $c, C > 0$, implying the claimed iteration bound. The same algorithm along with a conversion between the precision for geometric programming and the precision required for scaling problem (see Section 5.4.3) gives the following.

Corollary 5.2.3. *There is an algorithm that, given as input a well-conditioned instance of the scaling problem with shift (Problem 5.1.3), returns $x_\varepsilon \in \mathbb{R}^n$ such that*

$$\|\text{grad } F_\theta(x_\varepsilon)\|_2 = \|\text{grad } F(x_\varepsilon) - \theta\|_2 \leq \varepsilon$$

with the number of iterations bounded by

$$O\left(\sqrt{k} \log\left(k \frac{R_\theta}{r_\theta} \frac{R_\theta}{\varepsilon} \log(k\beta)\right)\right).$$

As an application, we note that Theorem 5.2.2 can be used to solve the weak membership problem for a convex polytope given in vertex-representation [GLS12]: upon input $\Omega \subseteq \mathbb{Q}^n$, $\theta \in \mathbb{Q}^n$, and $\varepsilon > 0$ (without assuming $\theta \in \text{conv}(\Omega)$), the weak membership problem asks to assert either that $d(\theta, \text{conv}(\Omega)) \leq \varepsilon$ or that $B(\theta, \varepsilon)$ is not contained in $\text{conv}(\Omega)$ (these conditions are not mutually exclusive). In order to decide this, one can run the algorithm from Corollary 5.2.6 with $q = (1, \dots, 1) \in \mathbb{R}^k$, and precision ε . If the algorithm does not terminate within the stated (polynomial) number of iterations, or if the returned point $x_\varepsilon \in \mathbb{R}^n$ does not satisfy $\|\text{grad } F(x_\varepsilon) - \theta\|_2 \leq \varepsilon$, one may conclude that $\theta \notin \text{conv}(\Omega)$, hence $B(\theta, \varepsilon)$ is not contained in $\text{conv}(\Omega)$ either. Otherwise, we obtain a point $x_\varepsilon \in \mathbb{R}^n$ such that $\|\text{grad } F(x_\varepsilon) - \theta\|_2 \leq \varepsilon$; since $\text{grad } F(x_\varepsilon) \in \text{conv}(\Omega)$, one can therefore safely assert that $d(\theta, \text{conv}(\Omega)) \leq \varepsilon$.

5.2.2. General instances

We now discuss our results for general instances (well-conditioned or not). Here we provide an interior-point algorithm that approximates the unconstrained GP to arbitrary precision with an iteration complexity bound that is *independent* of θ . For this, we prove a θ -independent diameter bound for approximate minimizers. The following quantity controls our bound (see Fig. 5.2).

Definition 5.2.4 (Facet gap). Let $\Omega \subseteq \mathbb{R}^n$ be a finite set. The *facet gap* $\varphi > 0$ of Ω is the smallest distance from any $\omega \in \Omega$ to the affine span of any facet of $\text{conv}(\Omega)$ not containing ω .

Note that the facet gap depends on Ω and not just on the Newton polytope.

Our diameter bound in terms of the facet gap (Theorem 5.3.3) generalizes a θ -independent diameter bound obtained in [SV19] for *integral* $\Omega \subseteq \mathbb{Z}^n$ to arbitrary

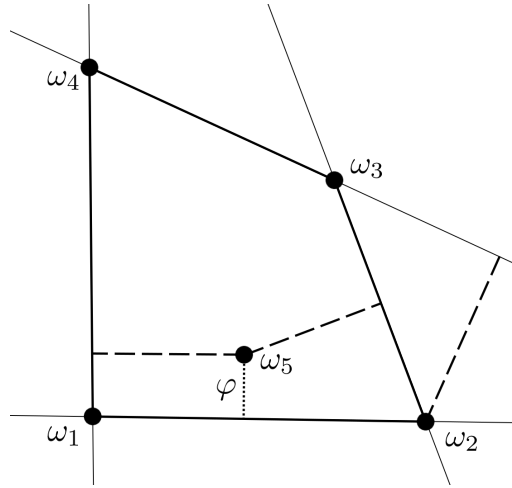


Figure 5.2.: The facet gap φ is the shortest line segment (dotted).

$\Omega \subseteq \mathbb{R}^n$, with only small modifications to the proof. The quantity that controls their diameter bound is called the *unary facet complexity* of the Newton polytope, denoted by ufc . We recover their diameter bound by showing that, in the integral case, the facet gap and the unary facet complexity are related by $\varphi^{-1} \leq \sqrt{n} \cdot \text{ufc}$; see Section 5.3.

We denote the *diameter* of the Newton polytope by

$$N = \max_{i \neq j} \|\omega_i - \omega_j\|_2. \quad (5.2.1)$$

Our algorithmic result is the following.

Theorem 5.2.5. *There is an interior-point algorithm (Algorithm 5.1) that, given as input an instance of the unconstrained GP problem (Problem 5.1.1) with shift $\theta \in \text{conv}(\Omega)$ and a lower bound $0 < \varphi_0 \leq \varphi$ on the facet gap, returns $x_\delta \in \mathbb{R}^n$ such that*

$$F_\theta(x_\delta) \leq F_\theta^* + \delta$$

within

$$O\left(\sqrt{k} \log\left(kn \frac{N}{\varphi_0} \frac{1}{\delta} \log\left(\frac{k\beta}{\delta}\right)\right)\right)$$

iterations. The starting point is determined explicitly by the input, and every iteration is a Newton step for a function that depends on φ_0 .

Theorem 5.2.5 applies to arbitrary points θ in the Newton polytope and achieves an iteration complexity that is fully independent of θ . In contrast, Theorem 5.2.2 applies only to well-conditioned instances and its complexity is sensitive to the distance of θ to the boundary of the Newton polytope. However, the former algorithm relies crucially on an a priori lower bound on the facet gap of Ω , while the latter has no such requirement. As such, our two algorithmic results are incomparable.

As in the well-conditioned case, our algorithm also allows one to solve the scaling problem with a similar iteration complexity bound.

Corollary 5.2.6. *There is an algorithm that, given as input an instance of the scaling problem (Problem 5.1.3) with shift $\theta \in \text{conv}(\Omega)$, as well as a lower bound $0 < \varphi_0 \leq \varphi$ on the facet gap, returns $x_\varepsilon \in \mathbb{R}^n$ such that*

$$\|\text{grad } F_\theta(x_\varepsilon)\|_2 = \|\text{grad } F(x_\varepsilon) - \theta\|_2 \leq \varepsilon$$

with the number of iterations bounded by

$$O\left(\sqrt{k} \log\left(kn \frac{N}{\varphi_0} \frac{R_\theta}{\varepsilon} \log\left(\frac{k\beta R_\theta}{\varepsilon}\right)\right)\right).$$

5.2.3. Condition measures for rational instances

Up to now, instances of the GP and scaling problems were allowed to be given by arbitrary real vectors. We now discuss how our condition measures (and thereby the iteration complexity) can for *rational* instances be effectively bounded in terms of the encoding length. We will focus our attention on r_θ and φ since the other condition measures β , R_θ , N can be straightforwardly bounded. Throughout, we follow the conventions of [GLS12] for the encoding length: we encode rational numbers (and rational vectors) in binary, and write $\langle \cdot \rangle$ for the encoding length.

By standard techniques, we derive in Section 5.5.1 polynomial upper bounds on $\log_2 r_\theta^{-1}$ and the facet gap in terms of the input bit-size. This implies that the iteration complexities of our interior-point methods are bounded by a polynomial in the encoding length of the instance.

We briefly compare the resulting guarantees to previous work. In the setting where $\Omega \subseteq \mathbb{Z}^n$ is integral, [BFG+19] gave a first-order method which solves the scaling problem in $\text{poly}(1/\varepsilon, R_\theta)$ iterations. They also developed a second-order method for the unconstrained GP problem based on the recently introduced notion of robustness whose iteration complexity is $\text{poly}(\log(1/\delta), R_\theta/r_\theta, n)$, see [ALOW17; CMTV17; CKV20]. Our results therefore improve upon both, as we have a logarithmic dependence on $1/\varepsilon$ (for the scaling problem) and $1/\delta$ (for the geometric program), and logarithmic dependence on R_θ/r_θ . An application of the ellipsoid method gives similar guarantees [NR99], but is usually not practical.

5.2.4. Total unimodularity

The general bounds on the condition measures in terms of the encoding length can be improved under a combinatorial hypothesis, as we show in Section 5.5.2. We call an instance *totally unimodular* if the exponents ω_i are all integral and the matrix A whose columns are given by the ω_i is totally unimodular, i.e., every subdeterminant is ± 1 or 0. Important examples of totally unimodular matrices are provided by the incidence matrices of directed graphs [Sch98, §19.3, Example 2]. We show that in this situation, the facet gap is bounded as $\varphi^{-1} \leq n^{3/2}$, see Theorem 5.5.4.² This implies that the interior-point algorithm in Theorem 5.2.5 can solve the unconstrained GP problem in $\tilde{O}(\sqrt{k} \log(\frac{1}{\varepsilon}))$ iterations and the scaling problem in $\tilde{O}(\sqrt{k} \log(\frac{1}{\varepsilon}))$ iterations. The $\tilde{O}(\cdot)$ notation here hides a polylogarithmic dependence on the input length.

²In [BFG+19], the similar bound $r_\theta^{-1} \leq 2^{(\theta)} n^{3/2}$ appears.

Many widely studied applications fall into this setting, among them matrix scaling; see Section 1.1. In this case, the underlying graph is a complete bipartite directed graph. Thus our interior-point algorithm runs in $\tilde{O}(\sqrt{k} \log(\frac{1}{\varepsilon}))$ iterations for finding an ε -approximate (r, c) -scaling of a non-negative matrix with k non-zero entries (if such a scaling exists). The matrix balancing problem can similarly be modeled by taking the underlying graph to be a complete directed graph. Unconstrained GPs arising from directed graphs can in general be related to nonlinear flow problems on directed graphs [CMTV17].

The iteration complexity that we obtain for matrix scaling and balancing slightly improves over (but is essentially the same as) the one given in [CMTV17] for an interior-point method designed specifically for these problems. It is natural to ask whether we can also meet the time complexity of the latter, which relied on a slightly different objective function and a clever implementation of approximate Newton iterations by using Laplacian solvers. We leave this question for future investigation.

5.3. Condition measures and diameter bounds

Throughout, we fix an instance of the unconstrained GP or scaling problem with $\Omega = \{\omega_1, \dots, \omega_k\} \subseteq \mathbb{R}^n$, $q \in \mathbb{R}_{>0}^k$, and shift $\theta \in \text{conv}(\Omega)$. Let W denote the direction vector space of the affine span $\text{aff}(\Omega)$, which equals the linear span of the $\omega_i - \theta$ in \mathbb{R}^n . Since the objective function $F_\theta: \mathbb{R}^n \rightarrow \mathbb{R}$ from Eq. (5.1.4) satisfies $F_\theta(x) = F_\theta(x + h)$ for all $h \in W^\perp$, we can restrict the optimization problem to W .

5.3.1. Well-conditioned instances

We first show the following diameter bound for a well-conditioned geometric program.

Proposition 5.3.1 (Well-conditioned diameter bound). *Assume $\theta \in \text{relint}(\text{conv}(\Omega))$. Then there exists $x \in W$ with $\|x\|_2 \leq \frac{\log \beta}{r_\theta}$ such that $F_\theta(x) = F_\theta^*$.*

Proof. For any $x \in W$ with $\|x\|_2 = 1$, $\max_{u \in \text{conv}(\Omega)} \langle u - \theta, x \rangle$ is the distance from θ to the face of $\text{conv}(\Omega)$ determined by the vector x . Minimizing over all such x results in the shortest distance from θ to any face of the polytope $\text{conv}(\Omega)$, which equals r_θ . This shows that

$$r_\theta = \min_{\substack{x \in W \\ \|x\|_2=1}} \max_{u \in \text{conv}(\Omega)} \langle u - \theta, x \rangle = \min_{x \in W \setminus \{0\}} \max_{i \in [k]} \frac{\langle \omega_i - \theta, x \rangle}{\|x\|_2}. \quad (5.3.1)$$

Therefore, if $x \in W$ satisfies $\|x\|_2 > \frac{\log(\beta)}{r_\theta}$, there exists $i_0 \in [k]$ such that $\langle \omega_{i_0} - \theta, x \rangle > \log(\beta)$. This shows that

$$e^{F_\theta(x)} \geq q_{i_0} e^{\langle \omega_{i_0} - \theta, x \rangle} > q_{i_0} \beta \geq \|q\|_1 = e^{F_\theta(0)}, \quad (5.3.2)$$

hence $F_\theta(x) > F_\theta(0)$. This completes the proof. \square

In the special case where $\Omega \subseteq \{0, 1\}^n$ and $q = (1, \dots, 1)$, this reduces to the diameter bound $\|x\|_2 \leq \frac{n}{r_\theta}$ shown in [SV14] and improved in [SV19] to $\|x\|_2 \leq \frac{\log k}{r_\theta}$ for general Ω but the same q .

Remark 5.3.2. Put $a_i = \omega_i - \theta$ and consider the matrix A with columns $\hat{a}_i := a_i / \|a_i\|_2$. The GCC condition number [BC13, §6.7] of A can be characterized as the inverse of

$$\min_{x \in W \setminus \{0\}} \max_{i \in [k]} \frac{\langle \hat{a}_i, x \rangle}{\|x\|_2}.$$

Upon replacing \hat{a}_i by a_i , this quantity becomes r_θ , see Eq. (5.3.1). The geometric condition measure r_θ thus is closely related to the GCC condition number (or Goffin measure), which is widely used in the context of testing polyhedral cone feasibility.

5.3.2. General instances

In this subsection we allow θ to be an arbitrary point in the Newton polytope. Here, the central quantity is the *facet gap* of Ω (Definition 5.2.4).

The following theorem improves upon [SV19, Thm. 4.1]. Its proof follows essentially the same argument, with a slight modification that also avoids the recursion and leads to a slightly better bound.

Theorem 5.3.3 (Diameter bound via facet gap). *For any $0 < \delta < 2\beta$ and $\theta \in \text{conv}(\Omega)$, there exists $x \in W$ such that*

$$\|x\|_2 \leq \frac{m}{\varphi} \log \left(\frac{2\beta}{\delta} \right)$$

and $F_\theta(x) \leq F_\theta^* + \delta$, where $m = \dim \text{aff}(\Omega) \leq n$.

Proof. To start, choose vectors $a_j \in W$ with $\|a_j\|_2 = 1$ and scalars $b_j \in \mathbb{R}$ for $j \in J$ some finite index set, such that the Newton polytope is defined by

$$\text{conv}(\Omega) = \{p \in \text{aff}(\Omega) : \langle p, a_j \rangle \leq b_j \quad \forall j \in J\}.$$

We assume each inequality defines a facet of the polytope. Define the *normal cone* N_ω at a vertex $\omega \in \Omega$ to be $N_\omega = \{\sum_{j \in J_\omega} c_j a_j : c_j \geq 0\}$, where $J_\omega = \{j \in J : \langle a_j, \omega \rangle = b_j\}$ is the set of tight constraints at ω . It is well-known that $W = \bigcup_{\omega \in \Omega} N_\omega$, where ω ranges over the vertices of $\text{conv}(\Omega)$ (the normal fan is complete).

Now fix $\theta \in \text{conv}(\Omega)$ and let $x^* \in W$ be such that

$$F_\theta(x^*) \leq F_\theta^* + \frac{\delta}{2}.$$

Then $x^* \in N_{\omega'}$ for some vertex $\omega' \in \Omega$ of $\text{conv}(\Omega)$, hence there exists a subset $J' \subseteq J_{\omega'} \subseteq J$ and non-negative numbers $\{c_j\}_{j \in J'}$ such that $x^* = \sum_{j \in J'} c_j a_j$. By Carathéodory's theorem, we may assume $|J'| \leq m = \dim W$. Now define

$$\Delta := \frac{1}{\varphi} \log \left(\frac{2\beta}{\delta} \right),$$

5. Interior-point methods for commutative scaling problems

which is positive by the assumption that $\delta < 2\beta$, and set

$$x := \sum_{j \in J'} \min(c_j, \Delta) a_j.$$

Since $\|a_j\|_2 = 1$, we have $\|x\|_2 \leq |J'| \cdot \Delta \leq m\Delta$, so x satisfies the desired norm bound.

To complete the proof, it therefore suffices to show that

$$F_\theta(x) \leq F_\theta(x^*) + \frac{\delta}{2}. \quad (5.3.3)$$

We start by setting $c'_j = \min(c_j, \Delta)$ for convenience, so that $x = \sum_{j \in J'} c'_j a_j$. Let J_0 consist of those $j \in J'$ such that $c'_j \neq c_j$, i.e., $c_j > \Delta$ and $c'_j = \Delta$. We may assume there exists at least one $j_0 \in J_0$; otherwise, $c_j \leq \Delta$ for every $j \in J'$, so we have $x = x^*$ and Eq. (5.3.3) holds trivially. Now consider the intersection of Ω with the face defined by the constraints J' ,

$$\Omega' = \{\omega \in \Omega : \langle a_j, \omega \rangle = b_j \quad \forall j \in J'\},$$

If $\omega \in \Omega \setminus \Omega'$, then $\omega \notin \text{aff}(\Omega')$, and

$$\begin{aligned} \langle \omega, x \rangle - \langle \omega', x \rangle &= \sum_{j \in J'} c'_j \langle \omega - \omega', a_j \rangle = \sum_{j \in J'} c'_j (\langle \omega, a_j \rangle - b_j) \\ &\leq c'_{j_0} (\langle \omega, a_{j_0} \rangle - b_{j_0}) = \Delta (\langle \omega, a_{j_0} \rangle - b_{j_0}) \\ &\leq -\varphi \Delta = \log \left(\frac{\delta}{2\beta} \right). \end{aligned}$$

The first inequality holds since each term in the sum is non-positive. The second inequality follows from the observation that the distance from ω to the affine span of the facet defined by a_{j_0} and b_{j_0} is

$$\frac{b_{j_0} - \langle \omega, a_{j_0} \rangle}{\|a_{j_0}\|_2} = b_{j_0} - \langle \omega, a_{j_0} \rangle \geq \varphi$$

by definition of the facet gap. So we obtain for $\omega \in \Omega \setminus \Omega'$ that

$$\beta e^{\langle \omega - \omega', x \rangle} \leq \frac{\delta}{2}. \quad (5.3.4)$$

On the other hand, if $\omega \in \Omega'$, then

$$\langle \omega - \theta, x \rangle = \langle \omega - \theta, x^* \rangle - \sum_{j \in J_0} (c_j - c'_j) \langle \omega - \theta, a_j \rangle \leq \langle \omega - \theta, x^* \rangle \quad (5.3.5)$$

since $c_j \geq c'_j$ and $\langle \theta, a_j \rangle \leq b_j = \langle \omega, a_j \rangle$ for all $j \in J'$. Therefore, we now obtain

$$\begin{aligned} F_\theta(x) &= \log \left(\sum_{i: \omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle} \right) + \log \left(1 + \frac{\sum_{i: \omega_i \notin \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle}}{\sum_{i: \omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle}} \right) \\ &\leq \log \left(\sum_{i: \omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle} \right) + \frac{\sum_{i: \omega_i \notin \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle}}{\sum_{i: \omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle}} \end{aligned}$$

$$\begin{aligned}
 &\leq \log \left(\sum_{i:\omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle} \right) + \frac{\sum_{i:\omega_i \notin \Omega'} q_i e^{\langle \omega_i - \theta, x \rangle}}{q_{i'} e^{\langle \omega' - \theta, x \rangle}} \\
 &\leq \log \left(\sum_{i:\omega_i \in \Omega'} q_i e^{\langle \omega_i - \theta, x^* \rangle} \right) + \frac{\delta}{2} \\
 &\leq F_\theta(x^*) + \frac{\delta}{2}.
 \end{aligned}$$

In the second inequality we denote by $i' \in [k]$ an index such that $\omega_{i'} = \omega'$, observing that $\omega' \in \Omega'$. The third inequality follows from Eqs. (5.3.4) and (5.3.5). \square

In contrast with the diameter bound for well-conditioned instances, which is in terms of the distance of θ to the boundary of the Newton polytope, the diameter bound in Theorem 5.3.3 is *independent* of the shift θ . However, the facet gap is not an intrinsic property of the Newton polytope $\text{conv}(\Omega)$, but depends on the entire set of exponents Ω , so the same is true for the diameter bounds in terms of the facet gap. The following example shows that this is necessary.

Example 5.3.4. Consider for $\varphi \in (0, 1/2]$ the instance with $\Omega = \{0, \varphi, 1\} \subseteq \mathbb{R}$, $q = (1, 1, 1) \in \mathbb{R}^3$, and $\theta = 0$. It is clear that $\text{conv}(\Omega) = [0, 1]$ and that Ω has facet gap equal to φ . Furthermore, we have

$$F_\theta(x) = \log(1 + e^{\varphi x} + e^x) \geq 0$$

and $\lim_{x \rightarrow -\infty} F_\theta(x) = 0$, so $F_\theta^* = 0$. On the other hand,

$$F_\theta(x) \geq \log(1 + e^{\varphi x}),$$

so any δ -approximate minimizer for $\delta \in (0, 1)$ must satisfy that $|x| \geq -x \geq \frac{1}{\varphi} \log \frac{1}{2\delta}$.

The following definition is from [SV19].

Definition 5.3.5 (Unary facet complexity). Let $P \subseteq \mathbb{R}^n$ be an integral polytope. The *unary facet complexity* $\text{ufc}(P)$ is the smallest integer $M \geq 0$ such that P can be described as the intersection of the affine span of P with half-spaces $\langle p, a \rangle \leq b$, where $a \in \mathbb{Z}^n$, $b \in \mathbb{R}$, and $\|a\|_\infty \leq M$.

We show now that the facet gap can be bounded in terms of the unary facet complexity.

Proposition 5.3.6. For $\Omega \subseteq \mathbb{Z}^n$ we have

$$\frac{1}{\varphi} \leq \sqrt{n} \cdot \text{ufc}(\text{conv}(\Omega)).$$

Proof. For any facet $F \subset \text{conv}(\Omega)$ there exists a corresponding half-space $\langle \cdot, a \rangle \leq b$ defined by $a \in \mathbb{Z}^n$, $b \in \mathbb{R}$, and $\|a\|_\infty \leq \text{ufc}(\text{conv}(\Omega))$. Then the affine span of the facet is given by $\text{aff}(F) = \text{aff}(\Omega) \cap H$, where H is the affine hyperplane

$$H = \{p \in \mathbb{R}^n : \langle a, p \rangle = b\}.$$

5. Interior-point methods for commutative scaling problems

As a consequence, the distance from any $\omega \in \Omega \setminus F$ to $\text{aff}(F)$ can be lower bounded by the distance of ω to the affine hyperplane H , that is,

$$d(\omega, \text{aff}(F)) \geq \frac{b - \langle a, \omega \rangle}{\|a\|_2} = \frac{\langle a, \omega' \rangle - \langle a, \omega \rangle}{\|a\|_2} \geq \frac{1}{\sqrt{n} \cdot \text{ufc}(\text{conv}(\Omega))},$$

where ω' is an arbitrary point in $\Omega \cap F$. To see the inequality, note that the numerator is positive and an integer since $a, \omega, \omega' \in \mathbb{Z}^n$, so at least 1, whereas the denominator is at most $\sqrt{n} \cdot \text{ufc}(\text{conv}(\Omega))$. \square

Thus, Theorem 5.3.3 and Proposition 5.3.6 imply the following diameter bound: for integral $\Omega \subseteq \mathbb{Z}^n$, there exists a δ -approximate minimizer of F_θ of norm

$$\|x\|_2 \leq n^{3/2} \text{ufc}(\text{conv}(\Omega)) \left(2L_p + \log \left(\frac{2k}{\delta} \right) \right),$$

where $L_p = \max_i |\log q_i|$ and we used that $\beta \leq k \frac{\max q_i}{\min q_i} \leq ke^{2L_p}$. The right-hand side bound is essentially the original diameter bound from [SV19] with a logarithmically improved dependence on n . The middle bound is very similar to a bound stated in an older version of [CKV20].

5.4. Interior-point methods for unconstrained geometric programming

In this section, we show that approximate minimizers of F_θ may be found efficiently using the interior-point method framework as in Chapter 4. The idea is to rewrite the geometric program as a *linear* optimization objective over a more complicated convex domain, for which we know an explicit *self-concordant barrier functional*. The domain and the corresponding barrier will be slightly different in the well-conditioned and the general case.

We first give the main ingredients that are common to the analysis of both the well-conditioned instances and the general instances. In the next two subsections we give the algorithms and complexity bounds for each case. Fix $\Omega = \{\omega_1, \dots, \omega_k\} \subseteq \mathbb{R}^n$, $q \in \mathbb{R}_{>0}^k$, and a shift $\theta \in \text{conv}(\Omega)$. Following the general strategy outlined above, we relate the geometric program to the minimization of a linear function over a compact convex domain. For $R > 0$, define

$$D_{\theta, R} = \left\{ (x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R} : \sum_{i=1}^k z_i \leq 1, \quad q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t \quad \forall i \in [k], \right. \\ \left. t \leq \log(5k\|q\|_1), \quad \|x\|_2 \leq R \right\}. \quad (5.4.1)$$

Here we recall that W is the span of the vectors $\omega_i - \theta$ or, equivalently, the direction vector space of $\text{aff}(\Omega)$. Note $(x, z, t) \in D_{\theta, R}$ implies $z_i > 0$ for all $i \in [k]$. The convexity of the domain $D_{\theta, R}$ follows from the convexity of the exponential map and of the ℓ^2 -norm ball.

Consider the linear objective $c = (0, \dots, 0; 0, \dots, 0; 1)$ on $D_{\theta, R}$. To see the relation between this objective and the unconstrained GP, note that for any $p = (x, z, t) \in D_{\theta, R}$, one has $\langle c, p \rangle = t$ and

$$F_{\theta}(x) = \log \sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle} \leq \log \sum_{i=1}^k z_i e^t \leq t, \quad (5.4.2)$$

so the minimum of the linear objective c on $D_{\theta, R}$ gives an upper bound on the minimum of the unconstrained GP restricted to the ball $\|x\|_2 \leq R$. In the following lemma we show that these minima are in fact the same. Consequently, if (x, z, t) is a δ -approximate minimizer of c on the domain $D_{\theta, R}$, then x is a δ -approximate minimizer of $F_{\theta}(x)$, restricted to vectors of norm $\|x\|_2 \leq R$.

Lemma 5.4.1 (Value). *For any $R > 0$, we have*

$$\text{val} := \min_{p \in D_{\theta, R}} \langle c, p \rangle = \min_{(x, z, t) \in D_{\theta, R}} t = \min_{\|x\|_2 \leq R} F_{\theta}(x), \quad (5.4.3)$$

$$V := \max_{p \in D_{\theta, R}} \langle c, p \rangle = \max_{(x, z, t) \in D_{\theta, R}} t = \log(5k\|q\|_1). \quad (5.4.4)$$

Furthermore, the difference $V - \text{val}$ satisfies

$$\log(5k) \leq V - \text{val} \leq \log(5k\beta) \quad (5.4.5)$$

Proof. For the first claim, note that Eq. (5.4.2) implies that

$$\text{val} \geq \min_{\|x\|_2 \leq R} F_{\theta}(x).$$

Now consider a minimizer x of the right-hand side, which we can assume to be in W . Then, $t := F_{\theta}(x)$ is such that

$$t \leq F_{\theta}(0) = \log\|q\|_1 \leq \log(5k\|q\|_1),$$

and if we set $z_i := q_i e^{\langle \omega_i - \theta, x \rangle - t}$ then

$$\sum_{i=1}^k z_i = \sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle} e^{-t} = e^{F_{\theta}(x)} e^{-F_{\theta}(x)} = 1.$$

Thus we find that $(x, z, t) \in D_{\theta, R}$, with $t = F_{\theta}(x)$, and Eq. (5.4.3) follows.

To see that Eq. (5.4.4) holds, note that the upper bound $V \leq \log(5k\|q\|_1)$ follows directly from the constraint on the t -variable, and this upper bound being an equality for the point

$$p = (0, \dots, 0; \frac{1}{k}, \dots, \frac{1}{k}; \log(5k\|q\|_1)) \in D_{\theta, R}.$$

Lastly, to show Eq. (5.4.5), note that

$$\text{val} = \inf_{\|x\|_2 \leq R} F_{\theta}(x) \geq \inf_{x \in \mathbb{R}^n} F_{\theta}(x) = F_{\theta}^* \geq \log \min_{i \in [k]} q_i$$

where the last inequality is Eq. (5.1.5). We clearly also have $\text{val} \leq F_{\theta}(0) = \log\|q\|_1$, so val satisfies

$$\log \min_{i \in [k]} q_i \leq \text{val} \leq \log\|q\|_1.$$

Combining this with Eq. (5.4.4) yields Eq. (5.4.5). \square

5. Interior-point methods for commutative scaling problems

The key to applying interior-point methods to unconstrained geometric programming is the following result, which gives an explicit barrier functional for $D_{\theta, R}$. It is well-known that such a barrier can be constructed, as it follows from standard barrier functionals (Section 4.2) and barrier combination rules.

Proposition 5.4.2 (Barrier). *The compact domain $D_{\theta, R} \subseteq W \times \mathbb{R}^k \times \mathbb{R}$ has non-empty interior. Moreover, it admits the self-concordant barrier functional*

$$\begin{aligned} \Psi_{\theta, R}(x, z, t) = & - \sum_{i=1}^k \log z_i - \sum_{i=1}^k \log (\log z_i - \langle \omega_i - \theta, x \rangle + t - \log q_i) \\ & - \log(\log(5k\|q\|_1) - t) - \log(1 - \sum_{i=1}^k z_i) - \log(R^2 - \|x\|_2^2), \end{aligned}$$

with complexity parameter $\nu = 2k + 3$.

Proof. It is clear that $D_{\theta, R}$ has non-empty interior (for example, Eq. (5.4.7) below gives a point in the interior). We now derive the barrier functional. It is well-known that the epigraph of the exponential, given by

$$\{(y, z) \in \mathbb{R} \times \mathbb{R} : e^y \leq z\},$$

admits the self-concordant barrier functional $(y, z) \mapsto -\log z - \log(\log z - y)$, with complexity parameter 2; see [NN94, Prop. 5.3.3]. Recall also that the logarithmic barrier functional $\tau \mapsto -\log \tau$ for the half line $\mathbb{R}_{\geq 0} \subseteq \mathbb{R}$ has complexity parameter 1. Then the closed convex set

$$\{(y, z, \tau) \in \mathbb{R}^k \times \mathbb{R}^k \times \mathbb{R} : e^{y_i} \leq z_i \text{ for all } i \in [k], \tau \geq 0\} \quad (5.4.6)$$

is simply the product of k copies of the epigraph of the exponential and the half line, so a barrier functional Ψ' is given by the sum of the barrier functionals for each term in the product [NN94, Prop. 2.3.1 (iii)], i.e.,

$$\Psi'(y, z, \tau) = - \sum_{i=1}^k \log z_i - \sum_{i=1}^k \log(\log z_i - y_i) - \log \tau.$$

The complexity parameter is then at most the sum of the individual complexity parameters, i.e., $2k + 1$. Next, note that

$$\{(x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R} : q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t \text{ for all } i \in [k], t \leq \log(5k\|q\|_1)\}$$

is the preimage of Eq. (5.4.6) under the injective affine transformation

$$\begin{aligned} A: W \times \mathbb{R}^k \times \mathbb{R} & \rightarrow \mathbb{R}^k \times \mathbb{R}^k \times \mathbb{R}, \quad (x, z, t) \mapsto (\langle \omega_1 - \theta, x \rangle - t + \log q_1, \dots \\ & \dots, \langle \omega_k - \theta, x \rangle - t + \log q_k; z_1, \dots, z_k; \log(5k\|q\|_1) - t), \end{aligned}$$

hence by [NN94, Prop. 2.3.1 (i)] admits the self-concordant barrier functional

$$\begin{aligned} (\Psi' \circ A)(x, z, t) = & - \sum_{i=1}^k \log z_i - \sum_{i=1}^k \log (\log z_i - \langle \omega_i - \theta, x \rangle + t - \log q_i) \\ & - \log(\log(5k\|q\|_1) - t) \end{aligned}$$

with the same complexity parameter $2k + 1$. Finally, we may incorporate the linear constraint $\sum_{i=1}^k z_i \leq 1$ by adding the logarithmic barrier $-\log(1 - \sum_{i=1}^k z_i)$, and the ℓ^2 -norm constraint $\|x\|_2 \leq R$ by adding the barrier $-\log(R^2 - \|x\|_2^2)$; see [Ren01, Prop. 2.3.1 (ii)]. This increases the complexity parameter to $2k + 3$ and results in the desired domain and barrier. \square

Next we need to bound the symmetry of a suitable starting point, as defined in Definition 4.3.2. For this, we will use the following lemma.

Lemma 5.4.3. *Let $D \subseteq E$ be a closed convex subset, and let $p \in D$. Suppose that $r < R$ are two radii such that $B(p, r) \subseteq D \subseteq B(p, R)$, where the closed balls are taken with respect to an arbitrary norm on E . Then, D is bounded, $p \in \text{int}(D)$, and*

$$\text{sym}(p) \geq \frac{r}{R}.$$

Proof. Clearly D is bounded and contains p in its interior. For the symmetry claim, note that for any $u \in D$, we have $u \in B(p, R)$, so $p - u \in B(0, R)$, and hence

$$p + \frac{r}{R}(p - u) \in B(p, r) \subseteq D.$$

This shows that $p + \frac{r}{R}(p - D) \subseteq D$, which implies the desired lower bound on the symmetry. \square

One important aspect of Lemma 5.4.3 is the freedom in choosing a norm; in particular, we do not assume that the norm comes from the inner product on E . We will now use this freedom to bound the symmetry of the following starting point:

$$p'_0 = (0, \dots, 0; \frac{1}{2k}, \dots, \frac{1}{2k}; \log(4k\|q\|_1)), \quad (5.4.7)$$

which is clearly contained in the interior of $D_{\theta, R}$.

Proposition 5.4.4 (Symmetry bound). *For any $R > 0$, we can bound the symmetry of $D_{\theta, R}$ with respect to the point p'_0 by*

$$\frac{1}{\text{sym}(p'_0)} \leq 10 \max(R_{\theta} R, k, \log(4k\beta)),$$

where we recall that $R_{\theta} = \max_{i \in [k]} \|\omega_i - \theta\|_2$.

Proof. We wish to apply Lemma 5.4.3 using the following norm on $W \times \mathbb{R}^k \times \mathbb{R}$:

$$\| (x, z, t) \| := \max \left\{ \frac{\|x\|_2}{R}, \frac{2}{3} \|z\|_{\infty}, \frac{|t|}{\log(4k\beta)} \right\}.$$

We first show that every $(x, z, t) \in D_{\theta, R}$ satisfies

$$\| (x, z, t) - p'_0 \| \leq 1. \quad (5.4.8)$$

5. Interior-point methods for commutative scaling problems

By definition $\|x\|_2 \leq R$. Moreover, $z_i > 0$, so

$$\frac{2}{3}|z_i - \frac{1}{2k}| \leq \frac{2}{3} \left(z_i + \frac{1}{2k} \right) \leq \frac{2}{3} \left(\sum_{j=1}^k z_j + \frac{1}{2k} \right) \leq \frac{2}{3} \left(1 + \frac{1}{2k} \right) \leq 1.$$

Moreover, by Eqs. (5.1.5), (5.4.3) and (5.4.4), it holds that

$$\log \min_{i \in [k]} q_i \leq F_\theta^* \leq \text{val} \leq t \leq V = \log(5k\|q\|_1),$$

hence

$$\begin{aligned} & \frac{|t - \log(4k\|q\|_1)|}{\log(4k\beta)} \\ & \leq \frac{\max\{\log(5k\|q\|_1) - \log(4k\|q\|_1), \log(4k\|q\|_1) - \log \min_{i \in [k]} q_i\}}{\log(4k\beta)} \\ & = \frac{\max\{\log \frac{5}{4}, \log(4k\beta)\}}{\log(4k\beta)} \leq 1. \end{aligned}$$

Thus we have proved Eq. (5.4.8).

We now show that $D_{\theta, R}$ contains any point $(x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R}$ in the ball

$$\|(x, z, t) - p'_0\| \leq \frac{1}{10 \max(R_\theta R, k, \log(4k\beta))} < \frac{1}{10}. \quad (5.4.9)$$

The latter implies that $\|x\|_2 \leq \frac{R}{10} \leq R$, so x certainly satisfies the norm bound. Moreover, Eq. (5.4.9) ensures that $\|x\|_2 \leq \frac{1}{10R_\theta}$, and so

$$q_i e^{\langle \omega_i - \theta, x \rangle} \leq q_i e^{R_\theta \|x\|_2} \leq q_i e^{1/10} \leq e^{1/10} \|q\|_1 \quad (5.4.10)$$

for all $i \in [k]$. Next, we also have $\frac{2}{3}|z_i - \frac{1}{2k}| \leq \frac{1}{10k}$, hence

$$\frac{7}{20k} \leq z_i \leq \frac{13}{20k}, \quad (5.4.11)$$

which implies that

$$\sum_{i=1}^k z_i \leq \frac{13}{20} \leq 1.$$

Finally, note that Eq. (5.4.9) entails

$$\frac{|t - \log(4k\|q\|_1)|}{\log(4k\beta)} \leq \frac{1}{10 \log(4k\beta)},$$

hence $|t - \log(4k\|q\|_1)| \leq \frac{1}{10}$, which implies that

$$\log(e^{-1/10} 4k\|q\|_1) \leq t \leq \log(e^{1/10} 4k\|q\|_1) \leq \log(5k\|q\|_1) \quad (5.4.12)$$

where the last inequality uses $4 \cdot e^{1/10} \leq 5$. This shows that $t \leq \log(5k\|q\|_1)$, which is necessary for $(x, z, t) \in D_{\theta, R}$. We now verify $z_i e^t \geq q_i e^{\langle \omega_i - \theta, x \rangle}$; combining the lower bound on t from Eq. (5.4.12) and the lower bound from Eq. (5.4.11) yields

$$z_i e^t \geq \frac{7}{20k} e^{-1/10} 4k\|q\|_1 = \frac{7}{5} e^{-1/10} \|q\|_1 \geq e^{1/10} \|q\|_1. \quad (5.4.13)$$

Together, Eqs. (5.4.10) and (5.4.13) show that $q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t$, as desired. Thus we have proved that $(x, z, t) \in D_{\theta, R}$ for any point in the satisfying Eq. (5.4.9). The bound on the symmetry then follows from Lemma 5.4.3, where the radius of the outer ball is given by Eq. (5.4.8) and the radius of the inner ball is given by Eq. (5.4.9). \square

In the remainder we consider two different situations. For general instances, we choose R according to a given lower bound on the facet gap, using Theorem 5.3.3. In the well-conditioned case, where θ is contained in the relative interior of the Newton polytope, we see that the upper bound on the z_i variables already leads to a bounded domain; this allows us to obtain an algorithm that is independent of any explicit radius bound.

5.4.1. General instances

Suppose the facet gap of the instance is lower bounded by some $\varphi_0 > 0$. Then, Theorem 5.3.3 and Eq. (5.4.3) show that for $\delta < 4\beta$ and

$$R = \frac{n}{\varphi_0} \log \left(\frac{2\beta}{\delta/2} \right)$$

the minimum value of $\langle c, (x, z, t) \rangle$ with $c = (0, \dots, 0; 0, \dots, 0; 1)$ and $(x, z, t) \in D_{\theta, R}$ is at most

$$\text{val} = \min_{\|x\|_2 \leq R} F_{\theta}(x) \leq F_{\theta}^* + \frac{\delta}{2}.$$

Therefore, in order to obtain a δ -approximate minimizer for the geometric program, it suffices to find a $\delta/2$ -approximate minimizer on $D_{\theta, R}$. The latter is achieved by Algorithm 5.1, which is an interior-point algorithm for the self-concordant barrier functional $\Psi_{\theta, R}$ derived above. Its iteration complexity is bounded by the following theorem.

Theorem 5.2.5. *There is an interior-point algorithm (Algorithm 5.1) that, given as input an instance of the unconstrained GP problem (Problem 5.1.1) with shift $\theta \in \text{conv}(\Omega)$ and a lower bound $0 < \varphi_0 \leq \varphi$ on the facet gap, returns $x_{\delta} \in \mathbb{R}^n$ such that*

$$F_{\theta}(x_{\delta}) \leq F_{\theta}^* + \delta$$

within

$$41\sqrt{k} \log \left(3600 k^2 n \frac{N}{\varphi_0} \frac{1}{\delta} \log^2 \left(\frac{5k\beta}{\delta} \right) \right) = O \left(\sqrt{k} \log \left(kn \frac{N}{\varphi_0} \frac{1}{\delta} \log \left(\frac{k\beta}{\delta} \right) \right) \right)$$

iterations. The starting point is determined explicitly by the input, and every iteration is a Newton step for a function that depends on φ_0 .

Algorithm 5.1: IPM for unconstrained GP: general case

Input: exponents $\omega_1, \dots, \omega_k \in \mathbb{R}^n$, coefficients $q \in \mathbb{R}_{>0}^k$, shift $\theta \in \text{conv}(\Omega)$, precision $0 < \delta < 1$, lower bound φ_0 on facet gap

Domain: $D_{\theta, R} = \{(x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R} : q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t \ \forall i \in [k], \sum_{i=1}^k z_i \leq 1, \|x\|_2 \leq R \text{ and } t \leq \log(5k\|q\|_1)\}$, where $W = \text{span}\{\omega_1 - \theta, \dots, \omega_k - \theta\}$, and $R = \frac{n}{\varphi_0} \log(\frac{2\beta}{\delta/2})$

Barrier: $\Psi_{\theta, R}(x, z, t) = -\log(R^2 - \|x\|_2^2) - \log(1 - \sum_{i=1}^k z_i) - \log(\log(5k\|q\|_1) - t) + \sum_{i=1}^k -\log z_i - \log(\log z_i - \langle \omega_i - \theta, x \rangle - \log q_i + t)$

Complexity parameter: $\nu = 2k + 3$

- 1 $p'_0 \leftarrow (0, \dots, 0; \frac{1}{2k}, \dots, \frac{1}{2k}; \log(4k\|q\|_1));$
 - 2 $c \leftarrow (0, \dots, 0; 0, \dots, 0; 1);$
 - 3 $(p_0, \eta_0) \leftarrow \text{PreliminaryStage}(p'_0, c);$
 - 4 $(x, z, t) \leftarrow \text{MainStage}(p_0, \eta_0, T = 10\sqrt{\nu} \log(\frac{6}{5} \frac{\nu}{\eta_0 \delta/2}), c);$
 - 5 **return** t
-

Proof. The result follows from applying Theorem 4.3.4 to find a $\frac{\delta}{2}$ -approximate minimizer, with the closed convex domain $D_{\theta, R}$, the self-concordant barrier functional $\Psi_{\theta, R}$ from Proposition 5.4.2 with complexity parameter $\nu = 2k + 3$, the symmetry bound given by Proposition 5.4.4, and the starting point Eq. (5.4.7), along with the estimate $\log(5k) \leq V - \text{val} \leq \log(5k\beta)$ from Eq. (5.4.5) and the bound $R_\theta = \max_i \|\omega_i - \theta\|_2 \leq N = \max_{i \neq j} \|\omega_i - \omega_j\|_2$, which holds for any $\theta \in \text{conv}(\Omega)$. The number of iterations is at most

$$\begin{aligned}
 & 18\sqrt{\nu} \log\left(\frac{36\nu}{\text{sym}(p'_0)} \frac{V - \text{val}}{\delta/2}\right) \\
 & \leq 41\sqrt{k} \log\left(3600k \frac{1}{\delta} \max\left(n \frac{R_\theta}{\varphi_0} \log\left(\frac{4\beta}{\delta}\right), k, \log(4k\beta)\right) \log(5k\beta)\right) \\
 & \leq 41\sqrt{k} \log\left(3600k^2 n \frac{N}{\varphi_0} \frac{1}{\delta} \log^2\left(\frac{5k\beta}{\delta}\right)\right) \\
 & = O\left(\sqrt{k} \log\left(kn \frac{N}{\varphi_0} \frac{1}{\delta} \log\left(\frac{k\beta}{\delta}\right)\right)\right). \quad \square
 \end{aligned}$$

If all the inputs for Algorithm 5.1 are rational and encoded in binary, then $\|q\|_1$, β , and φ_0 are at most exponentially large in the encoding length. Since the iteration complexity depends logarithmically (or even doubly logarithmically) on these quantities, the resulting iteration complexity is at most *polynomial* in the encoding length of the input. See Section 5.5 for details.

5.4.2. Well-conditioned instances

Now assume the instance is well-conditioned, so θ is contained in the relative interior of the Newton polytope. Here, we consider

$$D_\theta = \left\{ (x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R} : \sum_{i=1}^k z_i \leq 1, \ q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t \ \forall i \in [k], \right.$$

Algorithm 5.2: IPM for unconstrained GP: well-conditioned case

Input: exponents $\omega_1, \dots, \omega_k \in \mathbb{R}^n$, coefficient vector $q \in \mathbb{R}_{>0}^k$,
 shift $\theta \in \text{relint conv}(\Omega)$, precision $0 < \delta < 1$
Domain: $D_\theta = \{(x, z, t) \in W \times \mathbb{R}^k \times \mathbb{R} : q_i e^{\langle \omega_i - \theta, x \rangle} \leq z_i e^t \ \forall i \in [k],$
 $\sum_{i=1}^k z_i \leq 1 \text{ and } t \leq \log(5k\|q\|_1)\}$, where $W = \text{span}\{\omega_i - \theta\}$
Barrier: $\Psi_\theta(x, z, t) = -\log(1 - \sum_{i=1}^k z_i) - \log(\log(5k\|q\|_1) - t) +$
 $\sum_{i=1}^k -\log z_i - \log(\log z_i - \langle \omega_i - \theta, x \rangle - \log q_i + t)$
Complexity parameter: $\nu = 2k + 2$

```

1  p'_0 ← (0, ..., 0; 1/2k, ..., 1/2k; log(4k\|q\|_1));
2  c ← (0, ..., 0; 0, ..., 0; 1);
3  (p_0, η_0) ← PreliminaryStage(p'_0, c);
4  (x, z, t) ← MainStage(p_0, η_0, T = 10√ν log(6/5 * ν/η_0δ), c);
5  return t
    
```

$$t \leq \log(5k\|q\|_1)\},$$

which looks just like $D_{\theta, R}$ except that we omitted the norm bound on x . We claim that the two domains coincide for any

$$R \geq \frac{\log(5k\beta)}{r_\theta}. \quad (5.4.14)$$

Indeed, if there were some $(x, z, t) \in D_\theta$ with $\|x\|_2 > R$ then Eq. (5.3.2) would show that $q_{i_0} e^{\langle \omega_{i_0} - \theta, x \rangle} > 5k\|q\|_1$ for some $i_0 \in [k]$. This is a contradiction, since for any $(x, z, t) \in D_\theta$ we have

$$q_{i_0} e^{\langle \omega_{i_0} - \theta, x \rangle} \leq \sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle} \leq \sum_{i=1}^k z_i e^t \leq e^t \leq 5k\|q\|_1.$$

Thus we see that, indeed, $D_\theta = D_{\theta, R}$ for any R as in Eq. (5.4.14).

As a consequence, the value of the convex program for the domain D_θ is exactly equal to F_θ^* , as follows from Eq. (5.4.3). Moreover, the domain D_θ is bounded and satisfies the symmetry bound given in Proposition 5.4.4 with $R = \log(5k\beta)/r_\theta$. Since D_θ no longer depends explicitly on the radius bound, we can use the self-concordant barrier functional

$$\begin{aligned} \Psi_\theta(x, z, t) = & -\sum_{i=1}^k \log z_i - \sum_{i=1}^k \log(\log z_i - \langle \omega_i - \theta, x \rangle + t - \log q_i) \\ & - \log(\log(5k\|q\|_1) - t) - \log(1 - \sum_{i=1}^k z_i) \end{aligned} \quad (5.4.15)$$

with complexity parameter $\nu = 2k + 2$. Using this modification we readily obtain an interior-point algorithm for well-conditioned instances. Importantly, this algorithm does *not* explicitly depend on r_θ or any other condition measure. By contrast, Algorithm 5.1 required as input a lower bound on the facet gap. The algorithm is stated in Algorithm 5.2, and the following theorem gives a precise iteration bound.

Theorem 5.2.2. *There is an interior-point algorithm (Algorithm 5.2) that, given as input a well-conditioned instance of the unconstrained GP problem with shift (Problem 5.1.1), returns $x_\delta \in \mathbb{R}^n$ such that $F_\theta(x_\delta) \leq F_\theta^* + \delta$ within*

$$36\sqrt{k} \log \left(1440k^2 \frac{R_\theta}{r_\theta} \frac{1}{\delta} \log^2(5k\beta) \right) = O \left(\sqrt{k} \log \left(k \frac{R_\theta}{r_\theta} \frac{1}{\delta} \log(k\beta) \right) \right)$$

iterations. The starting point of the algorithm is determined explicitly by the input, and every iteration is a Newton step.

Proof. Apply Theorem 4.3.4 to find a δ -approximate minimizer, with the closed convex domain D_θ , the self-concordant barrier functional Ψ_θ given in Eq. (5.4.15) with complexity parameter $\nu = 2k + 2$, the symmetry bound given in Proposition 5.4.4 with $R = \log(5k\beta)/r_\theta$, and the starting point Eq. (5.4.7), along with the estimate on $(V - \text{val})$ from Eq. (5.4.5). The number of iterations is then at most

$$\begin{aligned} & 18\sqrt{\nu} \log \left(\frac{36\nu}{\text{sym}(p'_0)} \frac{V - \text{val}}{\delta} \right) \\ & \leq 36\sqrt{k} \log \left(144k \frac{1}{\text{sym}(p'_0)} \frac{\log(5k\beta)}{\delta} \right) \\ & \leq 36\sqrt{k} \log \left(144k^2 \frac{1}{\delta} 10 \log(5k\beta) \frac{R_\theta}{r_\theta} \log(5k\beta) \right) \\ & = 36\sqrt{k} \log \left(1440k^2 \frac{R_\theta}{r_\theta} \frac{1}{\delta} \log^2(5k\beta) \right) \\ & = O \left(\sqrt{k} \log \left(k \frac{R_\theta}{r_\theta} \frac{1}{\delta} \log(k\beta) \right) \right). \quad \square \end{aligned}$$

As in the situation of Theorem 5.2.5, if all the inputs in Algorithm 5.2 are rational, then the iteration complexity is again at most polynomial in the encoding length of the inputs. Again see Section 5.5 for details.

5.4.3. Geometric programming and scaling

In this subsection, we show that in order to solve the scaling problem with precision $\varepsilon > 0$, it suffices to solve the corresponding unconstrained geometric program with some precision $\delta = \delta(\varepsilon)$. This is a special case of the (easy direction of) quantitative version of the Kempf–Ness theorem stated in Theorem 2.6.7, and is well-known (see e.g. [SV19; BFG+19]), but re-stated and included for concreteness.

Lemma 5.4.5 (Smoothness). *For any $\omega_1, \dots, \omega_k, \theta \in \mathbb{R}^n$ and $q \in \mathbb{R}_{>0}^k$, the function*

$$F_\theta: \mathbb{R}^n \rightarrow \mathbb{R}, \quad F_\theta(x) = \log \sum_{i=1}^k q_i e^{(\omega_i - \theta, x)}$$

is L -smooth with $L = R_\theta^2$, where $R_\theta = \max_i \|\omega_i - \theta\|_2$. Recall that this means that its gradient is L -Lipschitz or, equivalently, that its Hessian has eigenvalues $\leq L$.

Proof. The gradient $\nabla F_\theta(x) \in \mathbb{R}^n$ is given by

$$\nabla F_\theta(x) = \frac{\sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle} (\omega_i - \theta)}{\sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle}}.$$

Therefore, the Hessian of F at x is the linear map $\nabla^2 F_\theta(x): \mathbb{R}^n \rightarrow \mathbb{R}^n$ given by

$$\nabla^2 F_\theta(x) = \frac{\sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle} (\omega_i - \theta)(\omega_i - \theta)^\top}{\sum_{i=1}^k q_i e^{\langle \omega_i - \theta, x \rangle}} - (\nabla F_\theta(x))(\nabla F_\theta(x))^\top.$$

Hence we see that the eigenvalues of the Hessian can be upper bounded by the eigenvalues of $M := \nabla^2 F_\theta(x) + (\nabla F_\theta(x))(\nabla F_\theta(x))^\top$, because $(\nabla F_\theta(x))(\nabla F_\theta(x))^\top$ is positive semidefinite. The matrix M is a convex combination of the rank-one matrices $(\omega_i - \theta)(\omega_i - \theta)^\top$, so we can bound its eigenvalues by R_θ^2 . \square

The following proposition then shows that the scaling problem can be solved by solving the corresponding geometric program with sufficient precision.

Proposition 5.4.6 (Scaling from optimization). *Assume that $\theta \in \text{conv}(\Omega)$, and let $x \in \mathbb{R}^n$ be such that $F_\theta(x) \leq F_\theta^* + \delta$ for some $\delta > 0$. Then,*

$$\frac{\|\text{grad } F_\theta(x)\|_2^2}{2R_\theta^2} \leq \delta.$$

In particular, to solve the scaling problem with precision $\varepsilon > 0$ it suffices to find a solution for the unconstrained GP with accuracy $\delta = \varepsilon^2/(2R_\theta^2)$.

Proof. A standard argument shows that an L -smooth function can always be decreased in controlled way by following a gradient step. Namely, if we define $x' = x - \frac{1}{L} \nabla F_\theta(x)$ then, using Taylor's expansion to second order and bounding the quadratic contribution using smoothness,

$$F_\theta(x') - F_\theta(x) \leq -\frac{1}{L} \|\text{grad } F_\theta(x)\|_2^2 + \frac{1}{2L} \|\text{grad } F_\theta(x)\|_2^2 = -\frac{1}{2L} \|\text{grad } F_\theta(x)\|_2^2.$$

As x is a δ -approximate minimizer of F_θ , we must have

$$\frac{1}{2L} \|\text{grad } F_\theta(x)\|_2^2 \leq \delta.$$

The desired bound follows since we have $L = R_\theta^2$ by Lemma 5.4.5. \square

This proposition allows one to deduce Corollaries 5.2.3 and 5.2.6 directly from Theorems 5.2.2 and 5.2.5, respectively.

5.5. Bounds on condition measures

In this section, we give bounds on the condition measures from Section 5.3 for *rational* instances in terms of their binary encoding length. These bounds show that our interior-point algorithms have polynomial iteration complexity. We also explain how to obtain tighter estimates under a total unimodularity assumption on the Newton polytope. Throughout this section, we follow the conventions of [GLS12]: we encode rational numbers and vectors in binary, and write $\langle \cdot \rangle$ for the encoding length.

5.5.1. General bounds

We first give lower bounds on r_θ and φ , the distance of θ to the boundary of the Newton polytope and the facet gap of Ω , respectively. All other condition measures can be directly bounded in terms of the input length.

Lemma 5.5.1. *Let $\Omega \subseteq \mathbb{Q}^n$. If $\theta \in \mathbb{Q}^n \cap \text{relint conv}(\Omega)$, then*

$$\log_2 \frac{1}{r_\theta} \leq 6n^2 \max_{i \in [k]} \langle \omega_i \rangle + \langle \theta \rangle - n.$$

If $\theta = 0$, this can be improved to $3n^2 \max_{i \in [k]} \langle \omega_i \rangle - n$. Moreover, we have $\log_2 \frac{1}{\varphi} \leq (6n^2 + 1) \max_{i \in [k]} \langle \omega_i \rangle - n$.

Proof. The polytope $\text{conv}(\Omega)$ has vertex complexity at most $v := \max_{i \in [k]} \langle \omega_i \rangle$, so by [GLS12, Lem. 6.2.4], it has facet complexity at most $\phi := 3n^2 v$. This means that the polytope can be defined by inequalities of the form $\langle \cdot, a \rangle \leq b$ for $a \in \mathbb{Q}^n$, $b \in \mathbb{Q}$ with encoding length $\langle a \rangle + \langle b \rangle \leq \phi$.

As a consequence, if F is any facet of $\text{conv}(\Omega)$ then its distance to θ can be lower bounded as

$$d(\theta, F) \geq d(\theta, \text{aff}(F)) \geq \frac{b - \langle \theta, a \rangle}{\|a\|_2}$$

for certain $a \in \mathbb{Q}^n$, $b \in \mathbb{Q}$ with $\langle a \rangle + \langle b \rangle \leq \phi$. Now we have $\|a\|_2 \leq 2^{\langle a \rangle - n}$ by [GLS12, Lem. 1.3.3], while $b - \langle \theta, a \rangle$ is a positive rational number with denominator of absolute value at most $2^{\langle a \rangle + \langle b \rangle + \langle \theta \rangle} \leq 2^{\phi + \langle \theta \rangle}$. We conclude that the distance from θ to the facet F is at least

$$\frac{b - \langle \theta, a \rangle}{\|a\|_2} \geq \frac{1}{2^{\phi + \langle \theta \rangle} 2^{\langle a \rangle - n}} \geq \frac{1}{2^{2\phi + \langle \theta \rangle - n}} = \frac{1}{2^{6n^2 \max_i \langle \omega_i \rangle + \langle \theta \rangle - n}}.$$

Since the facet was arbitrary this implies the first claim. If $\theta = 0$, then we instead estimate

$$\frac{b - \langle \theta, a \rangle}{\|a\|_2} = \frac{b}{\|a\|_2} \geq \frac{1}{2^{\langle b \rangle} 2^{\langle a \rangle - n}} \geq \frac{1}{2^{\phi - n}} \geq \frac{1}{2^{3n^2 \max_i \langle \omega_i \rangle - n}},$$

which proves the second claim.

The argument for the third claim is as for the proof of first claim, but with θ replaced by any ω_i not on the facet under consideration. \square

Finally, we show that the unary facet complexity of an integral polytope can be similarly bounded in terms of the encoding length. Via Proposition 5.3.6, this also implies a bound on the facet gap, albeit with a worse polynomial scaling in the dimension n .

Lemma 5.5.2. *Let $\Omega \subseteq \mathbb{Z}^n$. Then the unary facet complexity of $\text{conv}(\Omega)$ satisfies*

$$\log_2 \text{ufc}(\text{conv } \Omega) \leq 3n^3 \max_{i \in [k]} \langle \omega_i \rangle - n.$$

Proof. Again by [GLS12, Lem. 6.2.4], the polytope $\text{conv}(\Omega)$ may be described by inequalities of the form $\langle p, a \rangle \leq b$ with $a \in \mathbb{Q}^n$, $b \in \mathbb{Q}$ of total encoding length $\langle a \rangle + \langle b \rangle \leq \phi := 3n^2 \max_{i \in [k]} \langle \omega_i \rangle$. Multiplying by the denominators of a gives $a' \in \mathbb{Z}^n$, $b' \in \mathbb{Q}$ such that a' has encoding length at most $n\phi = 3n^3 \max_{i \in [k]} \langle \omega_i \rangle$. Now the desired inequality follows from the bound $\|a'\|_\infty \leq \|a'\|_2 \leq 2^{\langle a' \rangle - n}$. \square

5.5.2. Total unimodularity

We now show how to improve the bounds given above in case the set of exponents Ω satisfies a total unimodularity hypothesis.

Definition 5.5.3 (Total unimodularity). An integer matrix $A \in \mathbb{Z}^{n \times k}$ is called *totally unimodular* if every square submatrix of A has determinant 0, 1 or -1 . We say that $\Omega = \{\omega_1, \dots, \omega_k\} \subseteq \mathbb{Z}^n$ is *totally unimodular* if the associated matrix $A_\Omega = [\omega_1 | \dots | \omega_k]$ with columns $\omega_1, \dots, \omega_k$ is totally unimodular.

As an important source of totally unimodular instances, suppose that G is a directed graph with vertex set $V = [n]$, edge set E of size k , and edge weights $q_{ij} > 0$ for $ij \in E$. Since the incidence matrix of a directed graph is totally unimodular [Sch98, §19.3, Example 2], the associated geometric program

$$F_{G,\theta}(x) = \log \sum_{ij \in E} q_{ij} e^{x_i - x_j} - \langle \theta, x \rangle \quad (5.5.1)$$

is totally unimodular.

If Ω is totally unimodular, every ω_i has entries only in $\{\pm 1, 0\}$. Therefore, we can bound the radius of the smallest enclosed ball around any $\theta \in \text{conv}(\Omega)$, as well as the diameter of the Newton polytope by

$$R_\theta = \min_{i \in [k]} \|\omega_i - \theta\|_2 \leq N = \max_{i \neq j} \|\omega_i - \omega_j\|_2 \leq 2\sqrt{n}. \quad (5.5.2)$$

We now show that the inverse distance to the boundary and the inverse facet gap can similarly be upper bounded by a polynomial in n , which is an exponential improvement over the general bounds of Lemma 5.5.1.

Theorem 5.5.4 (Totally unimodular bounds). *Let $\Omega \subseteq \mathbb{Z}^n$ be totally unimodular. Then the unary facet complexity $\text{ufc}(\text{conv}(\Omega)) \leq n$. As a consequence, $\varphi \geq n^{-3/2}$. Furthermore, if $\theta \in \mathbb{Q}^n \cap \text{relint} \text{conv}(\Omega)$, then we have $r_\theta \geq 2^{-\langle \theta \rangle} n^{-3/2}$, which can be improved to $n^{-3/2}$ if $\theta = 0$.*

Proof. Assume first that $\text{conv}(\Omega)$ is a full-dimensional polytope. Then every facet of $\text{conv}(\Omega)$ is the convex hull of some affinely independent $v_1, \dots, v_n \in \Omega$. By Cramer's rule, the affine hyperplane spanned by the facet consists of all $x \in \mathbb{R}^n$ such that

$$\det \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & & & \\ \vdots & v_1 & \dots & v_n \\ x_n & & & \end{bmatrix} = 0.$$

Expanding the determinant along the first column gives the linear equation

$$\sum_{i=1}^n (-1)^i \det(D_i) x_i = -\det(D_0) \quad (5.5.3)$$

where D_i is obtained by deleting the $(i + 1)$ -th row from the matrix

$$D = \begin{bmatrix} 1 & \dots & 1 \\ v_{1,1} & \dots & v_{n,1} \\ \vdots & & \vdots \\ v_{1,n} & \dots & v_{n,n} \end{bmatrix}.$$

For $i, j \in [n]$, let D_i^j be obtained by deleting the first row and the j -th column from D_i ; then expanding the determinant $\det(D_i)$ along the first row gives

$$|\det(D_i)| \leq \sum_{j=1}^n |\det(D_i^j)| \leq n \quad (5.5.4)$$

since D_i^j is a submatrix of A_Ω and hence submodular. By replacing the equality in Eq. (5.5.3) by an inequality and varying over all facets, we obtain a complete set of defining inequalities for the polytope. This shows that the unary facet complexity $\text{ufc}(\text{conv}(\Omega))$ is at most n . The lower bound on the facet gap now follows from Proposition 5.3.6.

We now consider an arbitrary $\theta \in \mathbb{Q}^n$ in the interior of $\text{conv}(\Omega)$ and bound its distance to the boundary. Set $a = [\det(D_1), \dots, \det(D_n)]^\top$ and $b = -\det(D_0)$, so that the hyperplane defined by Eq. (5.5.3) reads $\langle x, a \rangle = b$. The distance from θ to the facet is then lower bounded by

$$\frac{|b - \langle \theta, a \rangle|}{\|a\|_2} \geq \frac{|b - \langle \theta, a \rangle|}{n^{3/2}},$$

where we used that $\|a\|_2 \leq \sqrt{n}\|a\|_\infty \leq n^{3/2}$ by Eq. (5.5.4). Note that $\langle \theta, a \rangle \neq b$, since θ is not contained in the hyperplane. Moreover, $a \in \mathbb{Z}^n$ and $b \in \mathbb{Z}$. Therefore, if $\theta = 0$ then $|b - \langle \theta, a \rangle| = |b|$ is an integer (in fact, equal to 1 by total unimodularity), while in general it is a rational number with denominator at most $2^{\langle \theta \rangle}$. In either case we obtain the desired lower bound on r_θ .

Finally, suppose that $\text{conv}(\Omega)$ has dimension $r < n$. Then there exists a set of vectors $U = \{u_1, \dots, u_{n-r}\}$ in $\{0, e_1, \dots, e_n\}$ such that $\text{conv}(\Omega \cup U)$ has dimension n . Moreover, $\Omega \cup U$ is still totally unimodular. Hence by the previous part of the proof, $\text{conv}(\Omega \cup U)$ has unary facet complexity at most n . Every facet of $\text{conv}(\Omega)$ is now the intersection of some facet of $\text{conv}(\Omega \cup U)$ with the affine span of Ω , so the unary facet complexity of $\text{conv} \Omega$ is also at most n . Furthermore, since the distance from θ to any facet of $\text{conv} \Omega$ is at least as large as the distance from θ to any facet of $\text{conv}(\Omega \cup U)$ not containing θ , we also inherit the lower bound on r_θ . \square

The lower bound $r_0^{-1} \geq n^{-3/2}$ when $\theta = 0$ already appears in [BFG+19, Cor. 6.11] as a lower bound on the *weight margin* $\gamma(\pi)$ of a representation $\pi : T(n) \rightarrow \text{GL}(V)$ whose weights are exactly Ω . The proof given there is similar to the one we give (as well as to the proof of [GLS12, Lem. 6.2.4], which is also a key ingredient for Lemma 5.5.1): both use Cramer's rule to express equations for facets of $\text{conv}(\Omega)$ in terms of subdeterminants of the matrix A_Ω , which are bounded by the total unimodularity.

The following corollary specializes Theorem 5.2.5 and Corollary 5.2.6 to the totally unimodular case, using Eq. (5.5.2) and the lower bound on the facet gap from Theorem 5.5.4.

Corollary 5.5.5. *There is an interior-point algorithm (Algorithm 5.1) that, given as input an instance of the unconstrained GP problem with shift and totally unimodular $\Omega \subseteq \mathbb{Z}^n$, returns $x_\delta \in \mathbb{R}^n$ such that $F_\theta(x_\delta) \leq F_\theta^* + \delta$ within*

$$O\left(\sqrt{k} \log\left(kn \frac{1}{\delta} \log\left(\frac{k\beta}{\delta}\right)\right)\right) = \tilde{O}\left(\sqrt{k} \log\left(\frac{1}{\delta}\right)\right)$$

iterations. Similarly, given an instance of the scaling problem with totally unimodular $\Omega \subseteq \mathbb{Z}^n$, the same algorithm returns $x_\varepsilon \in \mathbb{R}^n$ such that $\|\text{grad } F_\theta(x_\varepsilon)\|_2 \leq \varepsilon$ within

$$O\left(\sqrt{k} \log\left(kn \frac{1}{\varepsilon} \log\left(\frac{kn\beta}{\varepsilon}\right)\right)\right) = \tilde{O}\left(\sqrt{k} \log\left(\frac{1}{\varepsilon}\right)\right)$$

iterations. Here, the notation $\tilde{O}(\cdot)$ hides poly(input) terms inside the logarithm.

In particular, this theorem applies to matrix scaling (and balancing, see Chapter 13), by using the following geometric program which is totally unimodular:

$$F_\theta(x, y) = \log \sum_{i,j} q_{ij} e^{x_i - y_j - \langle r, x \rangle + \langle c, y \rangle}.$$

Here, even stronger bounds can be obtained: the diameter of the Newton polytope is $N = 2$ and the facet gap satisfies $\varphi \geq n^{-1/2}$, since the unary facet complexity of the Newton polytope is in fact equal to 1 [SV19].

For matrix scaling, the state of the art for general matrices is a near-linear time algorithm [CKL+22; BCK+23]. Prior to this work, the best was an interior-point method given in [CMTV17, Thm. 6.1], which obtains an iteration complexity of

$$\tilde{O}\left(\sqrt{k} \log\left(\frac{\|q\|_1}{\varepsilon}\right)\right) \tag{5.5.5}$$

to find an (r, c) -scaling of a nonnegative matrix. They use an objective that is slightly different from our F_θ , namely

$$\tilde{f}_\theta(x, y) = \sum_{i,j} q_{ij} e^{x_i - y_j} - \langle r, x \rangle + \langle c, y \rangle,$$

that is, the ‘shift’ is done additively instead of in the exponent. We see that the iteration complexity in Corollary 5.5.5 slightly improves over Eq. (5.5.5).

6. Preliminaries in Riemannian geometry

In this chapter, we recall some basic concepts in Riemannian geometry that we will need in the remainder of this part, and fix our notation. We mostly follow the conventions of [Lee18], and assume basic familiarity with the theory of smooth manifolds. See [Lee13; Lee18; BH13] for comprehensive introductions to differential geometry, Riemannian geometry and non-positive curvature, respectively.

6.1. Metric, lengths, distances

Throughout, we let M denote a connected Riemannian manifold. Unless specified otherwise, all differential geometric objects (manifolds, functions, sections, etc.) are assumed to be C^∞ -smooth. We write $T_p M$ and $T_p^* M$ for the tangent and cotangent space at a point $p \in M$, and write TM and T^*M the tangent and cotangent bundle of M , respectively. The space of sections of a vector bundle E on M is denoted by $\Gamma(E)$. Sections of the (co)tangent bundle are called (co)vector fields. Given a function f , we write df for its differential, which is a covector field. Then $Xf = df(X)$ is the directional derivative of f in direction X for any vector field X . The Lie bracket of two vector fields X and Y is the vector field $[X, Y]$ that acts as $[X, Y]f = X(Yf) - Y(Xf)$ on any function f . More generally, for $k, l \geq 0$, a (k, l) -tensor field is by definition a section of the bundle $T^{(k,l)}M := (TM)^{\otimes k} \otimes (T^*M)^{\otimes l}$ or, equivalently, a $C^\infty(M)$ -multilinear map $\Gamma(T^*M)^k \times \Gamma(TM)^l \rightarrow C^\infty(M)$; when $k = 1$ we can also think of it as a $C^\infty(M)$ -multilinear map $\Gamma(TM)^l \rightarrow \Gamma(TM)$.

The Riemannian metric on M is a smoothly varying family of inner products on the tangent spaces, i.e., for every $p \in M$ we have an inner product $\langle \cdot, \cdot \rangle_p$ on $T_p M$ such that the map $p \mapsto \langle \cdot, \cdot \rangle_p$ is a section of the bundle $T^{(0,2)}M$. The induced norm on $T_p M$ is denoted by $\|\cdot\|_p$. We write $\langle X, Y \rangle$ and $\|X\|$ for the functions computing the pointwise inner product and norm, respectively, of vector fields X, Y .

Using the Riemannian metric, we can define the *length* of a piecewise regular (meaning smooth and non-zero derivative) curve by $L(\gamma) = \int_a^b \|\dot{\gamma}(t)\|_{\gamma(t)} dt$. This is independent of the parameterization. In particular, we may always reparameterize such that the curve has unit speed, i.e., $\|\dot{\gamma}(t)\| = 1$, except for finitely many points; in this case the length is $L(\gamma) = b - a$. Given a notion of length, we define the *Riemannian distance* $d(p, q)$ between any two points $p, q \in M$ as the infimum of the lengths of all piecewise regular curves from p to q . In this way, M becomes a metric space. Its topology is the same as the original topology of the manifold M .

This chapter is adapted from [HNW23].

6.2. Covariant derivative and curvature

The Riemannian metric determines the *Levi-Civita connection* ∇ . It assigns to any two vector fields X and Y the *covariant derivative* $\nabla_X Y$ of Y along X , which is again a vector field, and is determined uniquely by being a connection on the tangent bundle (meaning it is C^∞ -linear in X , \mathbb{R} -linear in Y , and satisfies the product rule $\nabla_X(fY) = f\nabla_X Y + (Xf)Y$ for all functions f) which is *compatible with the metric* in the sense that $X\langle Y, Z \rangle = \langle \nabla_X Y, Z \rangle + \langle Y, \nabla_X Z \rangle$ and *symmetric* (torsion-free), meaning $\nabla_X Y - \nabla_Y X = [X, Y]$, where $[X, Y]$ denotes the Lie bracket. The $C^\infty(M)$ -linearity in X implies that $\nabla_X Y|_p$ depends only on the tangent vector $v := X_p$ at the point $p \in M$ and the values of Y in an arbitrarily small neighbourhood of p ; accordingly we will also write $\nabla_v Y$. Moreover, $X \mapsto \nabla_X Y$ defines a $(1,1)$ -tensor field, called the *total covariant derivative* ∇Y of Y .

One can uniquely extend the above to define connections and covariant derivatives for all tensor bundles $T^{(k,l)}M$ by demanding that for functions it agrees with the differential, that it satisfies a product rule with respect to tensor products, $\nabla_X(T \otimes S) = (\nabla_X T) \otimes S + T \otimes (\nabla_X S)$ for all vector fields X and tensor fields T, S , and that it commutes with all contractions. As a consequence,

$$\begin{aligned} X(T(\omega_1, \dots, \omega_k, Z_1, \dots, Z_l)) &= (\nabla_X T)(\omega_1, \dots, \omega_k, Z_1, \dots, Z_l) \\ &\quad + T(\nabla_X \omega_1, \omega_2, \dots, \omega_k, Z_1, \dots, Z_l) \\ &\quad + \dots + T(\omega_1, \dots, \omega_k, Z_1, \dots, Z_{l-1}, \nabla_X Z_l) \end{aligned} \quad (6.2.1)$$

for any (k, l) -tensor field T , vector fields X, Z_1, \dots, Z_l , and covector fields $\omega_1, \dots, \omega_k$. Again, we write $\nabla_v T := (\nabla_X T)_p$ as this only depends on the tangent vector $v := X_p$ at the point $p \in M$. For any (k, l) -tensor field T , the map $(\omega_1, \dots, \omega_k, X, Z_1, \dots, Z_l) \mapsto (\nabla_X T)(\omega_1, \dots, \omega_k, Z_1, \dots, Z_l)$ defines a $(k, 1+l)$ -tensor field, called the *total covariant derivative* and denoted by ∇T . We note that [Lee18] uses a different convention. In particular, we can define the *Hessian* of a function f as $\nabla^2 f = \nabla(\nabla f)$, which is a $(0, 2)$ -tensor field that turns out to be symmetric for the Levi-Civita connection; see Section 6.4.

Let $\tilde{M} \subseteq M$ be an embedded submanifold, equipped with the induced metric, and let $\tilde{\nabla}$ denote its Levi-Civita connection. If X, Y are vector fields on \tilde{M} that are extended arbitrarily to a neighborhood of \tilde{M} in M , then the *Gauss formula* holds on \tilde{M} :

$$\nabla_X Y = \tilde{\nabla}_X Y + \mathbb{I}(X, Y), \quad (6.2.2)$$

where $\mathbb{I}(X, Y) := \pi^\perp(\nabla_X Y)$ is the *shape tensor* or *second fundamental form* \mathbb{I} of \tilde{M} , with $\pi^\perp: TM|_{\tilde{M}} \rightarrow (T\tilde{M})^\perp$ the orthogonal projection [Lee18, Thm. 8.2].

While the covariant derivative itself is not a tensor field, it can be used to define the so-called *Riemann curvature tensor* which is a fundamental local invariant of Riemannian manifolds. Given vector fields X, Y, Z , we can define the vector field

$$R(X, Y)Z := \nabla_X(\nabla_Y Z) - \nabla_Y(\nabla_X Z) - \nabla_{[X, Y]}Z.$$

We may think of $R(X, Y)$ as a C^∞ -linear operator on the tangent bundle; hence R is a $(1, 3)$ -tensor field. The operator $R(X, Y)$ is skew-symmetric, and it is a skew-symmetric function of X and Y . It further satisfies the algebraic Bianchi identity $R(X, Y)Z + R(Y, Z)X + R(Z, X)Y = 0$. It can also be useful to define $R(X, Y, Z, W) := \langle R(X, Y)Z, W \rangle$, which is a $(0, 4)$ -tensor field.

A closely related object is the *sectional curvature*, which given two linearly independent tangent vectors $v, w \in T_p M$ at the same point $p \in M$ is defined by

$$K(v, w) = \frac{\langle R(v, w)w, v \rangle_p}{\langle v, v \rangle_p \langle w, w \rangle_p - \langle v, w \rangle_p^2}.$$

It only depends on the two-dimensional tangent plane spanned by v and w . The sectional curvature determines the Riemann curvature tensor uniquely. Its sign is an important characteristic of a Riemannian manifold. We say that M has *non-positive (sectional) curvature* if $K(v, w) \leq 0$ for all $v, w \in T_p M$ and $p \in M$.

One has the following useful geometric interpretation of sectional curvature [Mey89]:

Proposition 6.2.1. *Let $p \in M$ and $v, w \in T_p M$ be orthogonal unit vectors. Let $f(t) = \frac{1}{2}d(\text{Exp}_p(tv), \text{Exp}_p(tw))^2$. Then near $t = 0$, one has the Taylor expansion*

$$f(t) = t^2 - \frac{K(v, w)}{6}t^4 + O(t^5).$$

In other words, when the sectional curvature $K(v, w)$ is non-positive, the geodesics $\text{Exp}_p(tv)$ and $\text{Exp}_p(tw)$ diverge faster than one would expect from the situation in Euclidean space (see Section 6.3 for the definition of geodesics). There is also a more global interpretation of having non-positive curvature *everywhere* [BH13, Thm. 1.6]:

Theorem 6.2.2. *Let M be a simply connected complete Riemannian manifold. Then M has non-positive sectional curvature everywhere (i.e., $K(v, w)$ for all $p \in M$ and $u, v \in T_p M$) if and only if the CAT(0)-inequality¹ holds: for all $p, x, y, z \in M$ such that $d(x, y) = d(x, z) + d(z, y)$ (i.e., z is a midpoint of x and y), we have*

$$d(p, z)^2 \leq \frac{1}{2}d(p, x)^2 + \frac{1}{2}d(p, y)^2 - \frac{1}{4}d(x, y)^2. \quad (6.2.3)$$

We note that the Eq. (6.2.3) is tight for all $p, x, y, z \in M$ if and only if M has no curvature.

The next lemma records how these notions behave under rescaling of the Riemannian metric.

Lemma 6.2.3. *Let M be a Riemannian manifold with Riemannian metric $\langle \cdot, \cdot \rangle$, and let $c > 0$. Let M' be the same manifold but with Riemannian metric given by $\langle \cdot, \cdot \rangle' = c \langle \cdot, \cdot \rangle$. Then M' has the same Levi-Civita connection as M , and hence the same (1,3)-curvature tensor. For every $p, q \in M$, one has $d_{M'}(p, q) = \sqrt{c} d_M(p, q)$. Furthermore, for all $p \in M$ and linearly independent $v, w \in T_p M = T_p M'$, the sectional curvature satisfies $K_{M'}(v, w) = K_M(v, w)/c$.*

6.3. Parallel transport, geodesics, completeness

All definitions given so far restrict naturally to open subsets. However, it is often useful to restrict to curves in a manifold and differentiate a vector or tensor field

¹This inequality is named after Cartan, Alexandrov and Toponogov [BH13], and should not be confused with the $\mathfrak{K}(0)$ -inequality.

along it. If γ is a curve defined on an interval $I \subseteq \mathbb{R}$, then a (k, l) -tensor field along γ is a function $Y: I \rightarrow T^{(k,l)}M$ such that $Y(t) \in T_{\gamma(t)}^{(k,l)}M$ for every $t \in I$, i.e., a section of the pullback bundle $\gamma^*T^{(k,l)}$. Then there is a unique \mathbb{R} -linear operator D_t , called the *covariant derivative along γ* , that satisfies the product rule $D_t(fY) = \dot{f}Y + fD_tY$ for $f \in C^\infty(I)$ for $f \in C^\infty(I)$, and which agrees with $\nabla_{\dot{\gamma}(t)}$ for every tensor field that extends to a neighborhood of γ .

A vector or tensor field Y along a curve γ is called *parallel* if its covariant derivative along γ vanishes identically, i.e., $D_tY \equiv 0$. For any curve $\gamma: I \rightarrow M$, $0 \in I$, and any tensor $y_0 \in T_{\gamma(0)}^{(k,l)}M$, standard results in ordinary differential equations imply that there always exists a unique parallel tensor field Y along γ such that $\gamma(0) = y_0$, called the *parallel transport* of y_0 along γ . For any $t \in I$, we get a linear isomorphism $\tau_{\gamma,t}: T_{\gamma(0)}^{(k,l)}M \rightarrow T_{\gamma(t)}^{(k,l)}M$ by setting $\tau_{\gamma,t}(y_0) = Y(t)$ called a *parallel transport map*. This is useful to compute covariant derivatives: if T is a (k, l) -tensor field then for all $p \in M$, $v \in T_pM$, $\eta_1, \dots, \eta_k \in T_p^*M$, and $w_1, \dots, w_l \in T_pM$ we have

$$\nabla_v T(\eta_1, \dots, \eta_k, w_1, \dots, w_l) = \partial_{t=0} T_{\gamma(t)}(\tau_{\gamma,t}\eta_1, \dots, \tau_{\gamma,t}\eta_k, \tau_{\gamma,t}w_1, \dots, \tau_{\gamma,t}w_l), \quad (6.3.1)$$

where γ is an arbitrary curve such that $\gamma(0) = p$ and $\dot{\gamma}(0) = v$. We are often interested in parallel transport along the manifold's geodesics, which we introduce next.

A curve γ is called a *geodesic* if it is parallel to its own tangent vector field, i.e., $D_t\dot{\gamma} \equiv 0$. For every $p \in M$ and $v \in T_pM$, there is a unique geodesic $\gamma: I \rightarrow M$ with $\gamma(0) = p$ and $\dot{\gamma}(0) = v$, defined on some maximal open interval I containing 0. Note that $\dot{\gamma}(t) = \tau_{\gamma,t}(\dot{\gamma}(0))$ for all $t \in I$. If $1 \in I$, we define $\text{Exp}_p(v) := \gamma_v(1)$. We call M *geodesically complete* if $I = \mathbb{R}$, i.e., if geodesics with arbitrary initial data exist for arbitrary times. Then the exponential map is defined on the whole tangent space, $\text{Exp}_p: T_pM \rightarrow M$. The Hopf–Rinow theorem states that if M is connected, geodesic completeness is equivalent to completeness with respect to the Riemannian distance function, as well as to the Heine–Borel property (bounded closed subsets are compact).

Any length-minimizing curve is a geodesic when parameterized with unit speed. In general, geodesics are only locally length-minimizing, but when M is connected and complete then any two points $p, q \in M$ are connected by a length-minimizing geodesic, although there may be many other geodesics. However, if M is not only complete but also has non-positive sectional curvature, then by the Cartan–Hadamard theorem the exponential map at each point is a covering map. In particular, if M also is simply connected, then the exponential map is a diffeomorphism, so there is a *unique* (up to reparameterization) geodesic connecting any two points p and q . We will denote the corresponding parallel transport by $\tau_{p \rightarrow q}$. Manifolds that are simply connected, geodesically complete, and have non-positive sectional curvature are called *Hadamard manifolds*. This includes a great variety of spaces of import in applications, such as Euclidean and hyperbolic spaces, the positive definite matrices, and other symmetric spaces with non-positive curvature (see Chapters 9 and 10).

6.4. Gradient and Hessian

Given a function $f: D \rightarrow \mathbb{R}$ defined on an open subset $D \subseteq M$, we define its *gradient* as the vector field $\text{grad}(f)$ that is dual to its differential. That is, for all vector fields X we have

$$\langle \text{grad}(f), X \rangle = df(X) = Xf.$$

The *Hessian* of f is defined as the second covariant derivative $\nabla^2 f = \nabla(\nabla f) = \nabla df$, which is a $(0, 2)$ -tensor field, that is, a smoothly varying family of bilinear forms. By definition and using Eq. (6.2.1), we have for any two vector fields X and Y that

$$(\nabla^2 f)(X, Y) = (\nabla_X df)(Y) = X(df(Y)) - df(\nabla_X Y) = X(Yf) - (\nabla_X Y)f, \quad (6.4.1)$$

which implies that Hessian is a *symmetric* tensor, by the symmetry of the Levi-Civita connection. Since the Hessian is a symmetric tensor, it is determined by the associated quadratic form. The latter can be conveniently calculated in terms of geodesics: for any $p \in M$ and $v \in T_p M$,

$$(\nabla^2 f)_p(v, v) = \partial_{t=0}^2 f(\text{Exp}_p(tv)). \quad (6.4.2)$$

Using metric compatibility, one can write $(\nabla^2 f)(X, Y) = \langle \nabla_X \text{grad}(f), Y \rangle$, which shows that the $(1, 1)$ -tensor field $\text{Hess}(f) := \nabla \text{grad}(f)$ is the natural operator definition of the Hessian.

One can similarly consider higher covariant derivatives, but these need no longer be symmetric as a consequence of the non-vanishing of the curvature tensor. In particular, the third covariant derivative is no longer captured by its diagonal $(\nabla^3 f)_p(v, v, v) = \partial_{t=0}^3 f(\text{Exp}_p(tv))$. This complicates the theory of self-concordance, as we will discuss in Section 8.1.

6.5. Convexity

Finally we recall here some basic notions of convexity on Riemannian manifolds. We first discuss convexity of subsets and then turn to convexity of functions. We assume that M is connected and geodesically complete, so that any two points are connected by a (length-minimizing) geodesic.

A subset $D \subseteq M$ is called (*totally*) *convex* if for every geodesic $\gamma: [0, 1] \rightarrow M$ with $\gamma(0) \in D$ and $\gamma(1) \in D$, it holds that $\gamma(t) \in D$ for all $t \in [0, 1]$. We remark that, in general, two points can be connected by more than one geodesic; accordingly there is more than one natural definition of convexity. We are primarily interested in applications to Hadamard spaces, where any two points are connected by a unique geodesic, just like in Euclidean space.

A (not necessarily continuous) function $f: D \rightarrow \mathbb{R}$ defined on a convex subset $D \subseteq M$ is called *convex* if for every geodesic $\gamma: [0, 1] \rightarrow M$ with $\gamma(0) \in D$ and $\gamma(1) \in D$, it holds that $f \circ \gamma: [0, 1] \rightarrow \mathbb{R}$ is convex. That is, f is convex along all geodesics in its domain. Equivalently, f is convex if and only if its epigraph

$$E_f = \{(p, t) \in D \times \mathbb{R} : f(p) \leq t\} \quad (6.5.1)$$

is a convex subset of $M \times \mathbb{R}$. If the epigraph is also closed as a subset of $M \times \mathbb{R}$, then f is called *closed convex*. This useful condition controls the behavior of a convex function at its boundary, as in the following lemma, which thanks to the Hopf-Rinow theorem can be proved just like in the Euclidean case [Nes18, Thm. 3.1.4]. In particular, any *continuous* convex function on a *closed* domain is closed convex. Parts (i) and (ii) state that any closed convex function $f: D \rightarrow \mathbb{R}$ is lower semicontinuous, also if we extend it to M by setting $f(p) = \infty$ for $p \notin D$ (in fact, this characterizes when a convex function is closed, but we will not need this).

Lemma 6.5.1. *Let $f: D \rightarrow \mathbb{R}$ be a (not necessarily continuous) closed convex function defined on a convex subset $D \subseteq M$. Then:*

- (i) *If $(p_k) \subseteq D$ is a sequence s.t. $p_\infty := \lim_{k \rightarrow \infty} p_k \in D$, then $\liminf_{k \rightarrow \infty} f(p_k) \geq f(p_\infty)$.*
- (ii) *If $(p_k) \subseteq D$ is a sequence s.t. $\lim_{k \rightarrow \infty} p_k \notin D$, then $\lim_{k \rightarrow \infty} f(p_k) = \infty$.*
- (iii) *If for some $L \in \mathbb{R}$ the level set $\mathcal{L} = \{p \in D : f(p) \leq L\}$ is non-empty and bounded, then f attains its minimum.*

Proof. (i) We need to show: for any subsequence (p_{k_j}) such that $\lim_{j \rightarrow \infty} f(p_{k_j}) = f_\infty$ for some $f_\infty \in \mathbb{R} \cup \{\pm\infty\}$, we have that $f_\infty \geq f(p_\infty)$. If $f_\infty = \infty$ there is nothing to show. If $f_\infty \in \mathbb{R}$ then we have $\lim_{j \rightarrow \infty} (p_{k_j}, f(p_{k_j})) = (p_\infty, f_\infty) \in E_f$, since the epigraph is closed, and hence $f_\infty \geq f(p_\infty)$. Finally, we note $f_\infty = -\infty$ cannot occur. Indeed, if $f_\infty = -\infty$ then $f(p_{k_j}) \leq f(p_\infty) - 1$ for j large enough, hence $(p_{k_j}, f(p_\infty) - 1) \in E_f$ for j large enough and hence $\lim_{j \rightarrow \infty} (p_{k_j}, f(p_\infty) - 1) = (p_\infty, f(p_\infty) - 1) \in E_f$, which is a contradiction.

(ii) Assume this is not so. Then there are a subsequence (p_{k_j}) and $L \in \mathbb{R}$ such that $f(p_{k_j}) \leq L$ for all j . Now, $\lim_{j \rightarrow \infty} (p_{k_j}, L) = (p_\infty, L)$, where $p_\infty := \lim_{k \rightarrow \infty} p_k$, but each (p_{k_j}, L) is contained in the epigraph, and hence the same must be true for the limit. It follows that $p \in D$, which is a contradiction.

(iii) Since the level set \mathcal{L} is non-empty, it contains a sequence (p_k) such that $\lim_{k \rightarrow \infty} f(p_k) = f_* := \inf_{p \in D} f(p)$. Because the epigraph is a closed subset of $M \times \mathbb{R}$, the same is true for $\mathcal{L} \times \{L\} = E_f \cap (M \times \{L\})$, and hence \mathcal{L} is a closed subset of M . It is also bounded by assumption. By the Hopf-Rinow theorem, which is applicable because we assume that M is geodesically complete, it follows that \mathcal{L} is compact. After passing to a subsequence, we may therefore assume that $p_\infty := \lim_{k \rightarrow \infty} p_k$ exists and is in $\mathcal{L} \subseteq D$. For continuous f , we then have $f(p_\infty) = f_*$ and this concludes the proof. If f is not continuous then we can proceed as follows. First suppose that $f_* = -\infty$. Fix any $p_0 \in \mathcal{L}$. Because M is geodesically complete and \mathcal{L} is bounded, there exists a constant $C > 0$ such that we can write $p_k = \text{Exp}_{p_0}(u_k)$ for some $u_k \in T_{p_0}M$ such that $\|u_k\|_{p_0} = d(p_0, p_k) \leq C$ for all k . Then we can choose $\alpha_k \in (0, 1)$ such that $\alpha_k \rightarrow 0$ and $\alpha_k f(p_k) \rightarrow -\infty$. Then the points $q_k := \text{Exp}_{p_0}(\alpha_k u_k)$ satisfy

$$f(q_k) \leq (1 - \alpha_k)f(p_0) + \alpha_k f(p_k) = f(p_0) + \alpha_k(f(p_k) - f(p_0)) \rightarrow -\infty,$$

where the first inequality holds by geodesic convexity. In particular, there is some constant $K \in \mathbb{R}$ such that $f(q_k) \leq K < f(p_0)$ for large enough k .

Now, (q_k, K) is in the epigraph and converges to (p_0, K) , because $\alpha_k \rightarrow 0$ and $\|u_k\|_{p_0} \leq C$ for all k . But $f(p_0) > K$, so (p_0, K) is not in the epigraph. This contradicts the assumption that the epigraph is closed. Thus we must have that $f_* > -\infty$. Then, $\lim_{k \rightarrow \infty} (p_k, f(p_k)) = (p_\infty, f_*)$ and since the epigraph is closed, it must contain the latter, meaning that $f(p_\infty) \leq f_*$ and hence $f(p_\infty) = f_*$. \square

We will later (in Section 8.2) be in the situation that $D \subseteq M$ is open and we are interested in smooth objective functions $f: D \rightarrow \mathbb{R}$ that have a *closed convex extension*, meaning that f extends to a closed convex function on some convex superset of D . This is the case in particular if f extends to a continuous convex function on the closure \overline{D} .

Just like in the Euclidean setting [Nes18, Thm. 3.1.5], one can see that the sum of two closed convex functions is again closed convex.

Lemma 6.5.2. *Let $f_1: D_1 \rightarrow \mathbb{R}$, $f_2: D_2 \rightarrow \mathbb{R}$ be closed convex functions defined on convex subsets $D_1, D_2 \subseteq M$. Then the function $f_1 + f_2$ is a closed convex function on $D_1 \cap D_2$.*

Proof. It is clear that $f_1 + f_2$ is a convex function on $D := D_1 \cap D_2$. To see that it is closed, consider an arbitrary convergent sequence (p_k, t_k) in $E_{f_1+f_2}$, with limit point $(p_\infty, t_\infty) \in M \times \mathbb{R}$. By Lemma 6.5.1, since f_1 and f_2 are closed convex, we have

$$\liminf_{k \rightarrow \infty} f_1(p_k) \geq f_1(p_\infty) \quad \text{and} \quad \liminf_{k \rightarrow \infty} f_2(p_k) \geq f_2(p_\infty),$$

and hence

$$t_\infty = \lim_{k \rightarrow \infty} t_k \geq \liminf_{k \rightarrow \infty} f_1(p_k) + \liminf_{k \rightarrow \infty} f_2(p_k) \geq f_1(p_\infty) + f_2(p_\infty),$$

which means that $(p_\infty, t_\infty) \in E_{f_1+f_2}$. Hence $f_1 + f_2$ is closed. \square

As in the Euclidean setting, one can also characterize convexity differentially. In particular, a C^2 -smooth function $f: D \rightarrow \mathbb{R}$ defined on an open convex subset $D \subseteq M$ is convex if and only if the quadratic forms defined by the Hessian are positive semidefinite, i.e.,

$$(\nabla^2 f)_p(v, v) \geq 0 \tag{6.5.2}$$

for all $v \in T_p M$ and $p \in D$. We discuss two refinements of the notion of convexity (for simplicity only in the C^2 -smooth setting): If f is strictly convex along any geodesic in the domain, then f is called strictly convex. A sufficient condition for strict convexity is the following: for every $p \in D$, the Hessian $(\nabla^2 f)_p$ is positive definite, i.e., Eq. (6.5.2) holds with equality only for $v = 0 \in T_p M$. Similarly, we say that f is μ -strongly convex for some $\mu > 0$ if it is so along any unit-speed geodesic in the domain. This is the case if and only if, for all $v \in T_p M$ and $p \in D$,

$$(\nabla^2 f)_p(v, v) \geq \mu \|v\|_p^2.$$

In convex optimization, upper bounds on the Hessian of a convex function are often also useful. We say that f is ν -smooth (not to be confused with smoothness in

the sense of C^∞) if it is so along any unit-speed geodesic in the domain, that is, if and only if

$$(\nabla^2 f)_p(v, v) \leq \nu \|v\|_p^2$$

for all $v \in T_p M$ and $p \in D$. When M is a Hadamard space then it is well-known that the distance $d(\cdot, p_0)$ to any fixed point $p_0 \in M$ is convex, and that $\frac{1}{2}d^2(\cdot, p_0)$ is 1-strongly convex, just like in Euclidean space. However, the latter will in general no longer be smooth. We discuss these important functions in Chapter 9.

In this context, we also record the following two useful propositions, which are well-known, see e.g. [Udr94, Thm. 7.4.2]:

Proposition 6.5.3. *Let $f: D \rightarrow \mathbb{R}$ be a C^2 -smooth function defined on an open convex subset $D \subseteq M$, and assume that f is ν -smooth. Let $v = -\frac{1}{\nu} \text{grad}(f)_p$ and $q = \text{Exp}_p(v)$. If $q \in D$, then we have*

$$f(q) \leq f(p) - \frac{1}{2\nu} \|\text{grad}(f)_p\|_p^2.$$

Proof. Let $w \in T_p M$. Then function $g(t) = f(\text{Exp}_p(tw))$ satisfies

$$f(p) + t \, df_p(w) \leq g(t) = f(p) + t \, df_p(w) + \frac{1}{2} g''(t')$$

for some $t' \in [0, t]$ by the Lagrange remainder form of Taylor's theorem. From the ν -smoothness of f , it follows that g is also ν -smooth, and so $g''(t') \leq \nu \|w\|^2$. In particular, for $w = -\frac{v}{\nu} = -\frac{\text{grad}(f)_p}{\nu}$ we obtain

$$f(q) = g(1) \leq f(p) - \frac{1}{\nu} df_p(v) + \frac{1}{2} \|v\|_p^2 = f(p) - \frac{1}{2\nu} \|\text{grad}(f)_p\|_p^2. \quad \square$$

Proposition 6.5.4. *Let $f: D \rightarrow \mathbb{R}$ be a C^2 -smooth function defined on an open convex subset $D \subseteq M$, and assume that f is ν -smooth and μ -strongly-convex, and achieves its minimum at some $p_* \in D$. Let $v = -\frac{1}{\nu} \text{grad} f$ and $q = \text{Exp}_p(v)$. If $q \in D$, then we have*

$$f(q) - f(p_*) \leq \left(1 - \frac{\mu}{\nu}\right) (f(p) - f(p_*)).$$

Proof. We adapt the Euclidean proof of [BG19, Thm. 3.8]. If $w \in T_p M$ is such that $\text{Exp}_p(w) = p_*$, then by a similar argument as for Proposition 6.5.3,

$$f(p_*) \geq f(p) + df_p(w) + \frac{1}{2} \mu \|w\|_p^2.$$

Since $df_p(w) = \langle \text{grad}(f)_p, w \rangle$, we may use the inequality

$$\langle \text{grad}(f)_p, w \rangle = \left\langle \frac{1}{\sqrt{\mu}} \text{grad}(f)_p, \sqrt{\mu} w \right\rangle \leq \frac{1}{2\mu} \|\text{grad}(f)_p\|_p^2 + \frac{1}{2} \mu \|w\|_p^2$$

to conclude that $f(p_*) \geq f(p) - \frac{1}{2\mu} \|\text{grad}(f)_p\|_p^2$. Hence by Proposition 6.5.3

$$\begin{aligned} f(q) - f(p_*) &\leq f(p) - f(p_*) - \frac{1}{2\nu} \|\text{grad}(f)_p\|_p^2 \\ &= \left(1 - \frac{\mu}{\nu}\right) (f(p) - f(p_*)) + \frac{\mu}{\nu} \left(f(p) - f(p_*) - \frac{1}{2\mu} \|\text{grad}(f)_p\|_p^2 \right) \\ &\leq \left(1 - \frac{\mu}{\nu}\right) (f(p) - f(p_*)). \end{aligned} \quad \square$$

Let $D \subseteq M$ be a convex subset (not necessarily open) that is also an embedded submanifold. Equip D with the induced metric and let $\tilde{\nabla}$ denote its Levi-Civita connection. Then D is a totally geodesic submanifold, so its shape tensor \mathbb{I} vanishes [Lee18, Prop. 8.12]. Now let T be a $(0, l)$ -tensor field on D that is extended arbitrarily to a neighborhood of D in M . Then by Eqs. (6.2.1) and (6.2.2) we find that $\tilde{\nabla}T = \nabla T|_{(TD)^{\otimes(1+l)}}$, where the right-hand side notation means that we restrict ∇T to a $(0, 1 + l)$ -tensor field on D . In particular, we inductively see that for every function $f: M \rightarrow \mathbb{R}$ and every $l \geq 0$, the following holds on D :

$$\tilde{\nabla}^l \tilde{f} = \nabla^l f|_{(TD)^{\otimes l}}. \quad (6.5.3)$$

7. Interior-point methods on manifolds: overview

Interior-point methods have proven to be extremely successful in the context of convex optimization on Euclidean space, as explained in Chapter 4 and exemplified in the context of commutative scaling in Chapter 5. However, so far, these successes have been restricted to convex optimization on Euclidean space. While there is a strong connection between self-concordance-based interior-point methods and Riemannian geometry [Dui99; NT02; NN08], the framework of interior-point methods has not yet been generalized to objectives which are *geodesically convex*, i.e., convex on Riemannian manifolds. Indeed, while there have been previous attempts at extending interior-point methods to this setting [Udr97; Ji07; JMJ07], a satisfactory generalization of the Euclidean theory had still been elusive – in particular, the natural quadratic convergence analysis of Newton’s method for self-concordant functions, which in turn enables efficient path-following methods with global guarantees.

Instead, research on Riemannian optimization has so far largely focused on different approaches. There is extensive literature on first- and second-order methods for convex and non-convex optimization, see e.g. [Udr94; AMS09; Sat21; Bou23] for comprehensive overviews and [FS02; DPM03; ABM08; SH15; ZS16; AS20; WS22; SW22]. Recently, [LY22] gave a path-following method for non-convex constrained manifold optimization which does not use self-concordance. In another direction, geodesic updates can also be useful for Euclidean convex optimization problems [Per23a; Per23b].

We extend the interior-point method framework to Riemannian manifolds. We generalize the key notion of self-concordance, and show that (unlike prior definitions) it gives the same structural results and guarantees as in the Euclidean setting, in particular local quadratic convergence of Newton’s method. This allows us to give a path-following method for optimizing suitable objective functions over domains for which a self-concordant barrier is available, and we give complexity guarantees that match the Euclidean ones.

As explained in Chapter 1, we are particularly motivated to find efficient algorithms for the norm-minimization and scaling problems as defined in Section 2.6. However, the framework has applications beyond scaling problems. For instance, it allows us to answer: Given points p_1, \dots, p_m on a Riemannian manifold, what is the minimum radius ball that contains all these points? What is their geometric median, i.e., the point that minimizes the sum of distances to each p_i ? The first question has been studied before in the Riemannian setting [AN13; NH15], and [NH15] gave an algorithm for the specific case of hyperbolic space, yielding a ball with radius at most a factor $1 + \delta$ larger than the optimal radius in $O(1/\delta^2)$ iterations. The geometric median problem has been studied in [FVJ09; Yan10],

This chapter is adapted from [HNW23].

and [Yan10] gave an explicit subgradient algorithm on general manifolds, finding a point whose squared distance to the point achieving minimal sum of distances to the p_i is at most ε in $O(1/\varepsilon)$ iterations.

For the minimum-enclosing ball problem and the geometric median problem, our framework gives (to the best of our knowledge) the first algorithms for efficiently finding high-precision solutions in non-positive curvature. For the entire class of scaling and non-commutative optimization problems, our framework yields new algorithms that match the complexity guarantees of the state-of-the-art algorithms [BFG+19], while not obviously suffering from the same obstructions as those methods, opening up a new avenue for future research.

Indeed, the current state-of-the-art methods are fundamentally incapable of providing algorithms that run in polynomial time in all parameters for the general scaling problem. The main reason that we lack the kind of sophisticated optimization methods that are known in the Euclidean setting, as reviewed earlier in Section 1.3, is due to the geometry of the spaces that one has to optimize over, which poses fundamental new challenges and obstructions. The lack of a constructive analog of cutting-plane methods or the ellipsoid method [Rus19; CMB23], and the exponential volume growth of balls, form obstructions to efficient optimization. The latter can be used to prove black-box lower bounds for first-order algorithms, with a linear dependence on a bound to the distance of the optimizer of the objective [HM21a; CB22; CB23]. Moreover, the distance to an optimizer can be exponential in the input size in the context of scaling problems [FORW21].

To overcome these challenges and obstructions, it is natural to resort to methods which are capable of better exploiting the structure of the optimization problem at hand. Interior-point methods offer a powerful such framework in the Euclidean case, and they have already proved successful for commutative scaling problems (see Chapter 5). With this work, we hope to contribute a first clear step towards generalizing this powerful framework to the manifold setting.

We believe that our results suggest and reinforce several interesting directions for follow-up research. For instance, does every convex domain admit a self-concordant barrier, as is the case in the Euclidean setting? Do there exist self-concordant barriers with better barrier parameters which can be used for these applications, leading to better algorithms? Alternatively, can it be shown that our constructions are essentially optimal? Can interior-point methods on manifold always be initialized efficiently, and is there a suitable notion of duality?¹

In the remainder of this chapter, we give a more detailed overview of our results. We start with our proposed notion of self-concordance in Section 7.1, followed by a discussion of self-concordant barriers and a path-following method in Section 7.2. In Section 7.3 we give the first examples of self-concordant functions on manifolds, as well as examples of self-concordant barriers. In Section 7.4 we explain why the norm minimization and scaling problems as defined in Section 2.6 fit into this framework. The application to the minimum enclosing ball problem is discussed in Section 7.5, and we discuss geometric median problem in Section 7.6. We discuss future directions and open questions in more detail in Section 7.7.

¹The lack of nontrivial linear functions in the presence of curvature poses significant challenges.

7.1. Self-concordance and Newton's method on manifolds

Throughout, $f: D \rightarrow \mathbb{R}$ is a smooth function defined on a convex subset $D \subseteq M$ of a connected, geodesically complete Riemannian manifold M . Then f is called *convex* if it is convex along geodesics. Let ∇ denote the *covariant derivative* (or Levi-Civita connection), which allows taking derivatives of vector and tensor fields, and in particular to define Hessians $\nabla^2 f$ and higher derivatives (see Chapter 6). Then our proposed generalization of self-concordance to possibly curved manifolds is as follows.

Definition 7.1.1 (Self-concordance). For $\alpha > 0$, a convex function f is called α -*self-concordant* if, for all $p \in D$ and for all tangent vectors $u, v, w \in T_p M$, we have

$$|(\nabla^3 f)_p(u, v, w)| \leq \frac{2}{\sqrt{\alpha}} \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(v, v)} \sqrt{(\nabla^2 f)_p(w, w)}. \quad (7.1.1)$$

If f is closed convex, meaning its epigraph is closed, then f is called *strongly α -self-concordant*.

Self-concordance can be interpreted as giving a bound on the norm of the third derivative $(\nabla^3 f)_p$, that is, on the change of the Hessian $(\nabla^2 f)_p$, with respect to the (possibly degenerate) inner product defined by the Hessian itself. We say that f is α -*self-concordant along geodesics* if one requires the above bound only for $u = v = w$, that is, if for all $p \in D$ and for all $u \in T_p M$, we have

$$|(\nabla^3 f)_p(u, u, u)| \leq \frac{2}{\sqrt{\alpha}} ((\nabla^2 f)_p(u, u))^{3/2}. \quad (7.1.2)$$

When $M = \mathbb{R}^n$, the third derivative is a symmetric tensor and hence the two notions coincide. However, in general, the third derivative is *not* symmetric in all its arguments, and indeed its asymmetry is precisely related to the manifold's *curvature* via the Ricci identity [Lee18, Thm. 7.14], as we discuss in Section 8.1. Prior work only considered self-concordance along geodesics [Ji07] (which suffices for a damped Newton method) and did not take the asymmetry into account [Udr97; JM07].

Here we show explicitly that self-concordance is in general strictly stronger than self-concordance along geodesics (see Section 7.3), and it is the stronger notion that allows for the desired quadratic convergence of Newton's method – a cornerstone of the interior-point theory. Assume for simplicity that the Hessian $(\nabla^2 f)_p$ is positive definite for all $p \in D$. Then the *Newton iterate* of f at $p \in D$ is defined by minimizing the local quadratic approximation:

$$p_{f,+} := \text{Exp}_p(u^*), \quad u^* = \underset{u \in T_p M}{\text{argmin}} \left(f(p) + df_p(u) + \frac{1}{2} (\nabla^2 f)_p(u, u) \right).$$

The progress is quantified in terms of the *Newton decrement*, which is directly related to the gap between the original function value and the minimum of the local quadratic approximation. It is defined for any $\alpha > 0$ and $p \in D$ as

$$\lambda_{f,\alpha}(p) = \sup_{0 \neq u \in T_p M} \frac{|df_p(u)|}{\sqrt{\alpha (\nabla^2 f)_p(u, u)}}. \quad (7.1.3)$$

Then we prove following result on general Riemannian manifolds in Theorem 8.1.16:

Theorem 7.1.2 (Quadratic convergence). *Let $f: D \rightarrow \mathbb{R}$ be a strongly α -self-concordant function defined on an open convex set $D \subseteq M$, with positive definite Hessian. Let $p \in D$ be a point such that $\lambda_{f,\alpha}(p) < 1$. Then the Newton iterate remains in the domain, i.e., $p_{f,+} \in D$, and moreover*

$$\lambda_{f,\alpha}(p_{f,+}) \leq \left(\frac{\lambda_{f,\alpha}(p)}{1 - \lambda_{f,\alpha}(p)} \right)^2.$$

To relate the Newton decrements at p and $p_{f,+}$, we control the change in the Hessian of f along the geodesic from p to $p_{f,+}$. This crucially uses the notion of self-concordance of Eq. (7.1.1), rather than the weaker definition along geodesics as in Eq. (7.1.2). This is because there are two directions involved: the one of the geodesic, and the one corresponding to the subsequent Newton decrement.

7.2. Barriers and a path-following method on manifolds

Interior-point methods provide a natural and modular approach for minimizing an objective f constrained to a bounded convex domain $D \subseteq M$. We briefly recall the setup from Chapter 4. The key idea is to, rather than minimize f directly, minimize for $t > 0$ the function

$$F_t: D \rightarrow \mathbb{R}, \quad F_t := tf + F,$$

where F is a self-concordant “barrier” that is finite on D and diverges to ∞ on its boundary.² This automatically ensures the constraint, as F_t is finite only on D , and for large t the objective dominates. One then starts with an approximate minimizer of F and $t \approx 0$, and follows the *central path* $z(t) := \operatorname{argmin}_{p \in D} F_t(p)$ by iteratively performing two steps: increase t to some t' such that the current point is still not too far from $z(t')$, and then take a Newton step for $F_{t'}$ to move closer to it. For large enough $t > 0$, we arrive at an approximate minimizer of f on $D \subseteq M$.

More precisely, the function $F: D \rightarrow \mathbb{R}$ is required to be a (*non-degenerate strongly self-concordant*) barrier for D , with barrier parameter $\theta \geq 0$, which means that F is strongly 1-self-concordant, has positive definite Hessian, and $\lambda_F(p)^2 \leq \theta$ for all $p \in D$. The barrier parameter θ controls how rapidly t can be increased in every iteration.

In order to guarantee that Newton’s method indeed moves closer to the central path, we are interested in conditions on f that ensure that the functions F_t are self-concordant for every $t > 0$, with a constant independent of t . One way to guarantee this is to assume that the objective $f: D \rightarrow \mathbb{R}$ is *compatible* with the

²In the Euclidean setting, the barrier $F(x) = -\log x$ models the constraint that $x > 0$, and $F(X) = -\log \det X$ defines the constraint that X is a positive-definite matrix [NN94; Ren01]. Constraints are combined simply by adding the respective barriers. In the manifold setting, barriers are much harder to come by, but we give general constructions and concrete examples in Section 8.2 and Chapters 9 and 10.

barrier F in the following sense: there are constants $\beta_1, \beta_2 \geq 0$ such that, for all $p \in D$ and $u, v \in T_p M$,

$$\begin{aligned} |(\nabla^3 f)_p(u, v, v)| &\leq 2\beta_1 \sqrt{(\nabla^2 F)_p(u, u)} (\nabla^2 f)_p(v, v) \\ &\quad + 2\beta_2 \sqrt{(\nabla^2 F)_p(v, v)} \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(v, v)}. \end{aligned}$$

In particular, linear and quadratic functions are compatible with arbitrary self-concordant barriers, but these are not the only examples, and we crucially use this level of generality to give algorithms for the general scaling or non-commutative optimization problem. We expand on compatibility in Section 8.2.2, and show that it is also useful for constructing new self-concordant barriers, for instance for the epigraph of a function compatible with a self-concordant barrier (Theorem 8.2.11).³ Our notion of compatibility is inspired by a similar notion in the Euclidean setting, as is our analysis of the path-following method [NN94]. Its precise guarantees match those from the Euclidean setting, and are given in the following theorem, which we prove in Theorem 8.2.17:

Theorem 7.2.1 (Path-following method). *Let $D \subseteq M$ be an open, bounded, and convex domain, and let $f, F: D \rightarrow \mathbb{R}$ be smooth convex functions, such that F is a self-concordant barrier with barrier parameter $\theta \geq 0$ and f has a closed convex extension. Let $\alpha > 0$ be such that $F_t := tf + F$ is α -self-concordant for all $t \geq 0$. Let $p \in D$ be such that $\lambda_F(p) \leq \frac{\sqrt{\alpha}}{8}$, and let $\varepsilon > 0$. Then, using*

$$O\left(\left(1 + \sqrt{\frac{\theta}{\alpha}}\right) \log\left(\frac{(\theta + \alpha) \|df_p\|_{F,p}^*}{\varepsilon \sqrt{\alpha}}\right)\right)$$

Newton iterations, one can find a point $p_\varepsilon \in D$ such that

$$f(p_\varepsilon) - \inf_{q \in D} f(q) \leq \varepsilon.$$

The quantity $\|df_p\|_{F,p}^*$ is a lower bound on the variation $\sup_{q \in D} f(q) - \inf_{q \in D} f(q)$ of f over D (Lemma 8.2.18), and hence imposes a natural notion of scale in the complexity bound.

7.3. Examples of self-concordance: Squared distance in non-positive curvature

Self-concordance on manifolds is much more difficult to verify than for Euclidean space, and this begs the question whether nontrivial examples even exist. A natural candidate is $f(p) = d(p, p_0)^2$, the *squared distance* function to some point $p_0 \in M$. On Euclidean space, f is trivially self-concordant, as its third derivative

³While optimizing a function f on a domain D can always be reduced to optimizing a linear function over its epigraph $\{(p, t) \in D \times \mathbb{R} : f(p) < t\}$, this requires a barrier for the epigraph. We construct such a barrier precisely when f is compatible with F . However, it may be more difficult to initialize the path-following method on the epigraph rather than directly on D , so it can be advantageous to optimize f directly. See Section 10.1.

vanishes identically. In the presence of curvature the third derivative can be nonzero. Nevertheless, we prove that the squared distance is self-concordant on $\text{PD}(n)$ and, as a corollary, also on a broad class of manifolds with non-positive curvature.

We now discuss this in more detail. As in the introduction, we denote by $\text{PD}(n) = \text{PD}(n, \mathbb{C})$ the complex positive-definite matrices, endowed with the well-known *affine-invariant* Riemannian metric, which is given as follows. Since $\text{PD}(n)$ is an open subset of $\text{Herm}(n)$, the Hermitian $n \times n$ -matrices, we can identify the tangent space $T_P \text{PD}(n)$ at every $P \in \text{PD}(n)$ with $\text{Herm}(n)$. Then the Riemannian metric is defined as follows: for any two tangent vectors $U, V \in T_P \text{PD}(n)$, their inner product is

$$\langle U, V \rangle_P = \text{Tr} [P^{-1} U P^{-1} V].$$

With this metric, $\text{PD}(n)$ is a Hadamard manifold, i.e., a simply connected geodesically complete Riemannian manifold with non-positive curvature. Its geodesics, parallel transport, covariant derivatives, and so forth all have well-known closed-form expressions, which are amenable to tools from matrix analysis. For example, the geodesics through $P \in \text{PD}(n)$ are of the form $t \mapsto \sqrt{P} e^{tH} \sqrt{P}$ for $H \in \text{Herm}(n)$, and geodesic midpoints are the same as operator geometric means. The distance between two matrices $P, Q \in \text{PD}(n)$, defined as the minimum length of any path connecting them, is

$$d(P, Q) = \|\log(P^{-1/2} Q P^{-1/2})\|_{\text{HS}},$$

where $\|\cdot\|_{\text{HS}}$ denotes the Hilbert–Schmidt (i.e., Frobenius) norm. In Theorem 9.2.11 we show:

Theorem 7.3.1 (Self-concordance of squared distance). *For any $P_0 \in \text{PD}(n)$, the squared distance $f: \text{PD}(n) \rightarrow \mathbb{R}$ to P_0 , defined by $f(P) = d(P, P_0)^2$, is 2-self-concordant.*

We conjecture that the squared distance is actually 8-self-concordant, see Remark 9.2.10. Self-concordance on $\text{PD}(n, \mathbb{C})$ implies the same result for the squared distance on any convex subset of it. Therefore, the self-concordance holds on any Hadamard manifold that is also a so-called symmetric space;⁴ we will call this a *Hadamard symmetric space*. In particular, using [BH13, Prop. 10.58] we obtain the following result, which covers most non-positively curved spaces of import in applications, including the general scaling or non-commutative optimization problem (Section 10.1):

Corollary 7.3.2. *Let $G \subseteq \text{GL}(n, \mathbb{R})$ be an algebraic subgroup⁵ such that $g^T \in G$ for every $g \in G$. Set $M := \{g^T g : g \in G\} \subseteq \text{PD}(n, \mathbb{R})$. Then $M \subseteq \text{PD}(n, \mathbb{R})$ is a convex subset, and for every $p_0 \in M$, the function $f: M \rightarrow \mathbb{R}$, $f(p) = d(p, p_0)^2$ is 2-self-concordant.*

Hyperbolic space \mathbb{H}^n is a paradigmatic example of a manifold with non-positive curvature in this class. Corollary 7.3.2 implies that the squared distance function to a point in \mathbb{H}^n is 1-self-concordant, as one has to rescale the curvature by a factor 2

⁴Any such space is the product of a symmetric space of non-compact type and a Euclidean space [Hel79, Prop. V.4.2], and embeds, possibly after rescaling the metric on each of its de Rham factors, as a complete convex submanifold of $\text{PD}(n, \mathbb{R})$ for some $n \geq 1$, and hence also in $\text{PD}(n, \mathbb{C})$ [Ebe97, Thm. 2.6.5]. See [Hel79] for more background.

⁵This means that G is a subset of $\text{GL}(n, \mathbb{R})$ determined by polynomial equations in the matrix entries.

to obtain an isometric embedding into $\text{PD}(n, \mathbb{C})$. Similarly, the conjectured 8-self-concordance on $\text{PD}(n, \mathbb{C})$ would imply 4-self-concordance on \mathbb{H}^n .

We are able to prove the stronger result that the squared distance on \mathbb{H}^n is in fact 8-self-concordant, and that this is optimal, see Theorem 9.3.1. In contrast, the squared distance on hyperbolic space is $\frac{27}{2}$ -self-concordant along geodesics, as was shown previously in [Ji07, Lem. 11].⁶ It is an interesting open question whether there exists a universal constant $C > 0$ such that if M is a Hadamard manifold with all sectional curvatures in $[-\kappa, 0]$, then for every $p_0 \in M$, $f(p) = d(p, p_0)^2$ is C/κ -self-concordant.

Using the self-concordance of the squared distance, it is easy to construct a self-concordant barrier for its epigraph (Theorem 8.2.11). To this end we provide the following result, which applies in particular to $\text{PD}(n)$, hyperbolic space, and all other Hadamard symmetric spaces.

Theorem 7.3.3 (Epigraph barrier). *Let M be a Hadamard manifold, and let $p_0 \in M$. Assume that the function $f: M \rightarrow \mathbb{R}$, $f(p) = d(p, p_0)^2$ is α -self-concordant. Let $D = \{(p, S) \in M \times \mathbb{R} : f(p) < S\}$. Then, the function $F: D \rightarrow \mathbb{R}$ defined by*

$$F(p, S) = -\log\left(S - d(p, p_0)^2\right) + \frac{1}{\alpha} d(p, p_0)^2 \quad (7.3.1)$$

is strongly 1-self-concordant, and $\lambda_F(p, S)^2 \leq 1 + \frac{2}{\alpha} d(p, p_0)^2$.

The reason that the proposition does *not* state that F is a barrier is that the Newton decrement $\lambda_F(p, S)$ is not bounded by a constant, but rather depends on the distance to the point p_0 . To obtain a barrier, one needs to impose an additional constraint on the domain to force it to be bounded, for instance by requiring that $S < S_0$, which can be implemented by adding a logarithmic barrier term $-\log(S_0 - S)$ to F . The dependence of the Newton decrement on the distance to p_0 is caused by the term $\frac{1}{\alpha} d(p, p_0)^2$ in Eq. (7.3.1), but without this term the function would not be self-concordant. See also Theorem 8.2.14, where we construct a barrier for the sublevel set of a self-concordant function, with barrier parameter depending on the gap in function value.

We also provide a strengthening of the above theorem for hyperbolic space (see Theorem 9.3.7):

Theorem 7.3.4. *Let $M = \mathbb{H}^n$, $p_0 \in M$, and define $f: M \rightarrow \mathbb{R}$ by $f(p) = d(p, p_0)^2$. Let $D = \{(p, R, S) \in M \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} : RS - f(p) > 0\}$. Then the function $F: D \rightarrow \mathbb{R}$ by*

$$F(p, R, S) = -\log(RS - f(p)) + f(p)$$

is strongly $\frac{1}{2}$ -self-concordant. Furthermore, $\lambda_{F, \frac{1}{2}}(p, R, S)^2 \leq 4 + 4f(p)$.

The significance of this result is that it can be used to construct a barrier for the epigraph of the *distance* to a point, rather than the squared distance, by restricting to the subspace defined by the equation $S = R$. This is essential for applying the framework to the geometric median problem, see Section 7.6. In the Euclidean setting, the additional f -term is unnecessary; see for instance the proof

⁶They prove that $M_f = \sqrt{16/27}$, where the constant M_f is related to the constant α in our definition of self-concordance along geodesics by $M_f = 2/\sqrt{\alpha}$.

of [NN94, Prop. 5.4.3]. In our setting the proof is more complicated, as it involves a strengthening of the self-concordance estimate on the third derivative of the squared distance. The key estimates which enable our proof of the above theorem are given in Theorem 9.3.1.

7.4. Application I: Non-commutative optimization and scaling problems

Our first application is the one which motivated us to extend the framework in the first place. We briefly recap the setting from Section 2.6. Let $G \subseteq GL(n, \mathbb{C})$ be a connected algebraic subgroup such that $g^* \in G$ for all $g \in G$. Let $\pi: G \rightarrow GL(V)$ be a regular representation on a finite-dimensional complex vector space V . Assume V is endowed with an inner product such that the unitary matrices in G act unitarily. The general *norm minimization* problem asks to minimize the norm over the orbit of a given vector $v \in V$, that is, we wish to minimize $\|\pi(g)v\|$ over $g \in G$. Note that $\|\pi(g)v\|^2 = \langle v|\pi(g^*g)|v \rangle$ (we use here that $\pi(g)^* = \pi(g^*)$; this can be proven using the Cartan decomposition, see Section 10.1 for details). Accordingly, it suffices to minimize the function defined by⁷

$$\phi_v: M \rightarrow \mathbb{R}, \quad \phi_v(p) = \log \langle v|\pi(p)|v \rangle$$

over $M = \{g^*g : g \in G\} = G \cap PD(n)$. This function is convex along the geodesics of M . It is also $N(\pi)^2$ -smooth in the convexity sense, where $N(\pi)$ is the *weight norm* of the action, see Section 2.6 for details. Therefore, if ϕ_v is bounded from below, a simple gradient descent algorithm can be used to find a point $p \in M$ such that $\|\text{grad}(\phi_v)_p\| \leq \delta$ within $O(N(\pi)^2[\phi_v(I) - \inf_{q \in M} \phi_v(q)]/\delta^2)$ iterations, see Proposition 6.5.3 or [BFG+19, Thm. 4.2]. A more sophisticated box-constrained Newton method is able to find an ε -approximate minimizer p_ε of ϕ_v within $O((1 + R_0)N(\pi) \log[(\phi_v(I) - \inf_{q \in M} \phi_v(q))/\varepsilon])$ iterations, where $R_0 > 0$ is an upper bound on the distance to such a minimizer [BFG+19, Thms. 5.1 & 5.7]. Using our interior-point path-following method we prove the following result in Theorem 10.1.9:

Theorem 7.4.1 (Non-commutative optimization). *Let $0 \neq v \in V$ and $R_0, \varepsilon > 0$. Let $M = \{g^*g : g \in G\} \subseteq PD(n)$ and $D = \{p \in M : d(p, p_0) \leq R_0\}$, and define $\phi_v: M \rightarrow \mathbb{R}$ by $\phi_v(p) = \log \langle v|\pi(p)|v \rangle$. Then there is an algorithm that within $O((1 + R_0)N(\pi) \log(N(\pi)R_0/\varepsilon))$ iterations of the path-following method finds $p_\varepsilon \in D$ such that*

$$\phi_v(p_\varepsilon) - \inf_{p \in D} \phi_v(p) \leq \varepsilon.$$

This essentially matches the complexity of the box-constrained Newton method mentioned above, which is currently the state-of-the-art. There is a small difference, in that our complexity has $N(\pi)R_0$ in the logarithm, rather than the potential gap $\phi_v(I) - \inf_{q \in M} \phi_v(q)$; these are related since ϕ_v is $N(\pi)$ -Lipschitz. The

⁷This function differs from the Kempf–Ness function $F_v: G \rightarrow \mathbb{R}, g \mapsto \log\|g \cdot v\|$ defined in Chapter 2. Note that $\frac{1}{2}\phi_v(g^*g) = F_v(g)$. This changes certain estimates by factors of 2. The reason for the change in notation is that after identifying $K \backslash G$ with M via $Kg \mapsto g^*g$, working with ϕ_v is more pleasant.

approach we take to obtain this result is to use the barrier on M which arises from Corollary 7.3.2 and Theorem 7.3.3, and to show that ϕ_ν is compatible with the squared distance function, which is enough to implement the path-following method, as explained earlier. It would be very interesting to find a suitable barrier for this problem with a smaller barrier parameter (or prove that no such barrier exists).

7.5. Application II: Minimum-enclosing ball problem on PD(n)

Next we consider the *minimum enclosing ball (MEB)* problem: given distinct points $p_1, \dots, p_m \in M$, find $p \in M$ such that $R(p) := \max_i d(p, p_i)$ is minimal. When $M = \mathbb{R}^n$ is Euclidean space, this is a well-studied problem in computational geometry. There, it can be formulated as a second-order cone problem, to which interior-point methods are applicable (see, e.g., [KMY04]).

When M is a Hadamard manifold, the distance to a point is convex, and hence the MEB problem is a convex optimization problem. In particular, for hyperbolic space $M = \mathbb{H}^n$, there has been previous work on the MEB problem [AN13; NH15]. The only algorithm with explicit complexity bounds that we are aware of is due to Nielsen and Hadjeres [NH15]. If R_* is the minimal radius of an MEB and $\delta > 0$, then they can find a point $p \in \mathbb{H}^n$ such that $\max_i d(p, p_i) \leq (1 + \delta)R_*$ within $O(1/\delta^2)$ iterations of an algorithm, each of which is simple to implement.

To find MEBs using interior-point methods, it is sufficient to have a barrier for the epigraph of the squared distance. In particular, the barrier constructed using Theorems 7.3.1 and 7.3.3 can be used to solve this problem on PD(n), and we prove the following result in Theorem 10.2.5:

Theorem 7.5.1 (Minimum enclosing ball). *Let $p_1, \dots, p_m \in \text{PD}(n)$ be $m \geq 3$ points, and set $R_0 = \max_{i \neq j} d(p_i, p_j)$. Let $R(p) = \max_i d(p, p_i)$, set $R_* = \inf_{p \in M} R(p)$, and let $\varepsilon > 0$. Then with $O((m+1)R_0^2)$ iterations of a damped Newton method and*

$$O\left(\sqrt{1 + m(R_0^2 + 1)} \log\left(\frac{m(R_0^2 + 1)}{\varepsilon}\right)\right)$$

iterations of the path-following method, one can find $p_\varepsilon \in \text{PD}(n)$ such that

$$R(p_\varepsilon) - R_* \leq \varepsilon.$$

A similar result can be obtained on arbitrary Hadamard symmetric spaces. We also note that the optimal radius R_* satisfies $R_0 \leq 2R_*$ (Lemma 10.2.2), so that the above also yields a multiplicative error guarantee. Compared to the results of [NH15], we have a logarithmic dependence on the precision ε , but a linear dependence on R_0 (as opposed to no dependence).

7.6. Application III: Geometric median on hyperbolic space

Our last application is the *geometric median problem*. In the Euclidean setting this is also known as the *Fermat–Weber problem* [CLM+16]. It is formally defined as follows: given points $p_1, \dots, p_m \in M$, not all contained in a single geodesic, find $p_0 \in M$ such that

$$p_0 \in \operatorname{argmin}_{p \in \mathbb{H}^n} s(p) := \sum_{j=1}^m d(p, p_j).$$

The objective function s is convex on Hadamard manifolds M . In contrast with the *geometric mean (or barycenter) problem*, which is to find the minimizer of $\sum_{j=1}^m d(p, p_j)^2$, finding the geometric median is non-trivial even on $M = \mathbb{R}^n$. The first and one of the best-known algorithms for this problem on Euclidean space is Weiszfeld's algorithm [Wei37], which is a simple iterative procedure based on solving the first-order optimality condition $\operatorname{grad}(s)_p = \sum_{j=1}^m (p - p_j)/d(p, p_j) = 0$ for p , while treating the $d(p, p_j)$ as constants. Unfortunately, the update rule is not well-defined when p is one of the p_j 's (which can be fixed, see e.g. [Ost78]), and it may converge very slowly in general. In [XY97] it was observed that one can also apply interior-point methods, by viewing the geometric median problem as a second-order cone program. More recent work [CLM+16] has shown that a specialized long-step interior-point method is capable of solving the geometric median problem on \mathbb{R}^n in nearly-linear time, and we refer the reader to their paper for a broader literature review. Weiszfeld's approach has been generalized to the Riemannian setting [FVJ09]. A sub-gradient approach [Yan10] can find a point with squared distance to the minimizer of s at most ε in $O(1/\varepsilon)$ iterations; however, in the negatively curved setting, it suffers from an exponential dependence on the quantity $R_0 = \max_{i \neq j} d(p_i, p_j)$.

We can solve the geometric median problem on hyperbolic space \mathbb{H}^n by using our interior-point framework and our barrier for the epigraph of the *distance* constructed using Theorem 7.3.4, which serve as analogs of the second-order cone and the associated barrier. In Theorem 10.3.5 we prove:

Theorem 7.6.1 (Geometric median). *Let $p_1, \dots, p_m \in \mathbb{H}^n$ be $m \geq 3$ points, not all on one geodesic, and set $R_0 = \max_{i \neq j} d(p_i, p_j)$. Define $s: \mathbb{H}^n \rightarrow \mathbb{R}$ by $s(p) = \sum_{j=1}^m d(p, p_j)$, and let $\varepsilon > 0$. Then with $O((m+1)R_0^2)$ iterations of a damped Newton method and*

$$O\left(\sqrt{m(R_0^2 + 1)} \log\left(\frac{mR_0(R_0^2 + 1)}{\varepsilon}\right)\right)$$

iterations of the path-following method, one can find $p_\varepsilon \in \mathbb{H}^n$ such that

$$s(p_\varepsilon) - \inf_{q \in \mathbb{H}^n} s(q) \leq \varepsilon.$$

For not too small ε , the cost is dominated by the damped Newton method, which we use to find a good starting point for the path-following method. We leave it as an open problem as to whether this can be avoided. Furthermore, the above applies only to \mathbb{H}^n rather than to $\text{PD}(n)$: it relies on the barrier constructed

using Theorem 7.3.4, which uses a non-trivial strengthening of the self-concordance estimates for the squared distance. We expect that such a strengthening can also be obtained more generally, and this would immediately generalize the algorithmic result from Theorem 7.6.1 to these spaces; we also leave this as a problem for future work.

7.7. Outlook

We summarize the results in this section and mention some future directions. We extend the basic theory of interior-point methods to manifolds, and show that the developed framework is capable of capturing interesting geodesically convex optimization problems. In particular, we define a suitable version of self-concordance on Riemannian manifolds, and show that it gives the same guarantees for Newton’s method as in the Euclidean setting. This is used to analyze a path-following method for the optimization of compatible objectives over domains for which one has a self-concordant barrier. We exhibit non-trivial examples of self-concordant functions, namely squared distance functions on $PD(n)$, and more generally symmetric spaces with non-positive curvature, and construct related self-concordant barriers. The framework is able to capture the optimization of Kempf–Ness functions, a problem which has connections to many areas of mathematics and computer science, leading to algorithms with state-of-the-art complexity guarantees. It also applies to computing the geometric median on hyperbolic space, for which we give an algorithm capable of finding high-precision solutions. This demonstrates the power of the framework, and we believe that it encompasses many more problems.

Our work highlights known challenges and suggests new directions:

- It is natural to search for self-concordant barriers for the aforementioned applications which have better barrier parameters. Alternatively, is it possible to prove lower bounds that show that the constructions given in our work are essentially optimal?
- In Euclidean convex optimization, there are universal constructions of self-concordant barriers, cf. [NN94; Hil14; Fox15; BE19; Che23]. Can one find such a construction for manifolds? We describe a concrete proposal. Let $D \subseteq M$ be a compact convex subset of a Hadamard manifold M , with non-empty interior. Denote by CM^∞ the cone over the boundary at infinity of M [Hir22a]. Its elements can be identified with the geodesic rays γ emanating from a fixed base point and hence determine Busemann functions b_γ as in Eq. (10.1.6). Define $F^*: CM^\infty \rightarrow \mathbb{R}$ by $F^*(\gamma) = \log \int_D \exp(-b_\gamma(q)) \, d\text{vol}(q)$. Then the inverse Legendre–Fenchel conjugate $F: D \rightarrow \mathbb{R}$ of F^* , given by $F(p) = \sup_{\gamma \in CM^\infty} -b_\gamma(p) - F^*(\gamma)$, is a natural candidate for a barrier for D . Indeed, for Euclidean space $M = \mathbb{R}^n$ it reduces precisely to the *entropic barrier* of Bubeck and Eldan [BE19].
- From the perspective of interior-point methods, we currently only treat the *main stage*, which minimizes an objective given a starting point that is well-centered with respect to the barrier F . Can one give a general procedure for finding such a starting point from an arbitrary feasible point $p \in D$? In the

Euclidean setting, this is achieved by applying the path-following method with the linear objective $f := -\langle \text{grad}(F)_p, \cdot \rangle$ in reverse, starting at $t = 1$. This is sensible as p is exactly a minimizer of $F_t = tf + F$ at $t = 1$. Busemann functions generalize linear functions to Hadamard manifolds, hence is natural to instead use $f = b_\gamma$ with γ the geodesic ray starting at $p \in M$ with direction $\text{grad}(F)_p$. When f is compatible with F (as we show in Section 10.1 for specific f and F), then one can use the same time steps as for the main stage, and switch to the main stage as soon as $\lambda_{F,\alpha} \leq \frac{1}{3}$. One method for lower bounding the t for which this happens is as follows: if F is μ -strongly convex and f is ν -smooth, then $\lambda_{F,\alpha}(q)$ is at most $\lambda_{F_t,\alpha}(q)\sqrt{1 + t\nu/\mu} + t\|df_q\|_{F,q,\alpha}^*/\alpha$, and $\|df_q\|_{F,q,\alpha}^*$ can be bounded (for instance) using Lipschitzness of f and strong convexity of F . We leave a more careful analysis of this idea to future work. We note that in the Euclidean setting, the complexity is often bounded in terms of the *(a)symmetry* of domain D with respect to the point p , see Section 4.3 and [NN94, Eq. (3.2.24)] for details, but such a bound does not seem to generalize to the Riemannian setting.

- It would be interesting to understand whether there is a suitable notion of primal-dual methods in the Riemannian setting, or a notion of duality which interacts well with self-concordance. While there exists a version of Legendre–Fenchel duality for Hadamard manifolds M , where the dual space is CM^∞ , the cone over the boundary at infinity of M discussed above, the conjugate of a convex function need not be convex [Hir22a]. Other proposals such as [BHS+21] require a stronger notion of convexity.

8. Interior-point methods on manifolds: the framework

In this chapter we generalize the Euclidean (self-concordance based) interior-point method framework to the setting of Riemannian manifolds. In Section 8.1 we define self-concordance and show that it yields guarantees for Newton's method that are familiar from the Euclidean setting. In Section 8.2 we turn to self-concordant barriers, define a notion of compatibility of an objective with a self-concordant barrier, and show that for these objectives one can give a implement a path-following method.

8.1. Self-concordance and Newton's method

In this section we generalize the notion of self-concordance and the corresponding analysis of Newton's method from the Euclidean setting to the Riemannian setting, and we comment on complications incurred by curvature. For expositions of the Euclidean theory of self-concordance and interior-point methods we refer to [NN94; Nes18; Ren01]. Throughout this section we assume that M is a connected and geodesically complete Riemannian manifold.

8.1.1. Self-concordance

Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex subset $D \subseteq M$. Then the Hessian is positive semidefinite, by Eq. (6.5.2), hence induces a (semi-)norm at each point. The rate of change of the Hessian is captured by the third covariant derivative, $\nabla^3 f = \nabla(\nabla(\nabla f)) = \nabla(\nabla^2 f)$. A function is called self-concordant if the latter can be bounded in terms of the former, as follows:

Definition 8.1.1 (Self-concordance). Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex subset $D \subseteq M$, and let $\alpha > 0$. We say that f is α -self-concordant if, for all $p \in D$ and for all $u, v, w \in T_p M$, we have

$$|(\nabla^3 f)_p(u, v, w)| \leq \frac{2}{\sqrt{\alpha}} \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(v, v)} \sqrt{(\nabla^2 f)_p(w, w)}, \quad (8.1.1)$$

It is called *strongly α -self-concordant* if is not just convex but closed convex, that is, if its epigraph (6.5.1) is a closed subset of $M \times \mathbb{R}$.

Here we follow the conventions of [NN94]. To interpret the definition, let us for a convex function f , a point p in its domain, and $\alpha > 0$ define the positive

This chapter is adapted from [HNW23].

semidefinite bilinear form and seminorm

$$\langle v, w \rangle_{f,p,\alpha} = \frac{(\nabla^2 f)_p(v, w)}{\alpha} \quad \text{and} \quad \|u\|_{f,p,\alpha} = \sqrt{\frac{(\nabla^2 f)_p(u, u)}{\alpha}}. \quad (8.1.2)$$

When the Hessian is positive definite (as is the case, e.g., when f is strongly convex), these endow M with a new Riemannian metric. In convex optimization, $\langle \cdot, \cdot \rangle_{f,p,\alpha}$ is called the “local inner product” and $\|\cdot\|_{f,p,\alpha}$ the “local norm”, but we will refrain from using this terminology as it is ambiguous in the Riemannian setting. For $\alpha = 1$, we will usually abbreviate $\langle \cdot, \cdot \rangle_{f,p} := \langle \cdot, \cdot \rangle_{f,p,1}$ and $\|\cdot\|_{f,p} := \|\cdot\|_{f,p,1}$. We can now rewrite Eq. (8.1.1) as follows:

$$|(\nabla^3 f)_p(u, v, w)| \leq 2\alpha \|u\|_{f,p,\alpha} \|v\|_{f,p,\alpha} \|w\|_{f,p,\alpha}. \quad (8.1.3)$$

Thus self-concordance can be interpreted as a boundedness of the third covariant derivatives at each point with respect to the seminorms defined by the Hessian.

We record some basic properties. Recall that self-concordant functions are defined on an open and convex domain, by definition.

Lemma 8.1.2. (i) *Let f be a (strongly) α -self-concordant function and let $c > 0$. Then cf is (strongly) $c\alpha$ -self-concordant.*

(ii) *Let $f_k: D_k \rightarrow \mathbb{R}$ be α_k -self-concordant functions for $k = 1, 2$, and suppose $D := D_1 \cap D_2$ is non-empty. Then $f := f_1 + f_2: D \rightarrow \mathbb{R}$ is α -self-concordant, with $\alpha := \min(\alpha_1, \alpha_2)$. If the functions f_k are strongly α_k -self-concordant, then f is strongly α -self-concordant.*

(iii) *Let $f_k: D_k \rightarrow \mathbb{R}$ be α -self-concordant functions for $k = 1, 2$. Then the function $f: D_1 \times D_2 \rightarrow \mathbb{R}$ defined by $f(p_1, p_2) := f_1(p_1) + f_2(p_2)$ is α -self-concordant. If both functions f_k are strongly α -self-concordant, then so is f .*

Property (i) follows from the definition, and (iii) follows from (ii). Before we prove (ii), we give a simpler characterization of self-concordance. As the Hessian is symmetric, third covariant derivatives are symmetric in the last two arguments. This can also be seen explicitly from the following formula for the third covariant derivative $\nabla^3 f$, which follows from Eq. (6.2.1) and holds for any three vector fields X, Y, Z :

$$(\nabla^3 f)(X, Y, Z) = X\left((\nabla^2 f)(Y, Z)\right) - (\nabla^2 f)(\nabla_X Y, Z) - (\nabla^2 f)(Y, \nabla_X Z). \quad (8.1.4)$$

This leads to the following simplification:

Lemma 8.1.3. *A convex function $f: D \rightarrow \mathbb{R}$ defined on an open convex subset $D \subseteq M$ is α -self-concordant if, and only if, for all $p \in M$ and $u, v \in T_p M$, we have*

$$|(\nabla^3 f)_p(u, v, v)| \leq \frac{2}{\sqrt{\alpha}} \sqrt{(\nabla^2 f)_p(u, u)} (\nabla^2 f)_p(v, v) \quad (8.1.5)$$

or, equivalently,

$$|(\nabla^3 f)_p(u, v, v)| \leq 2\alpha \|u\|_{f,p,\alpha} \|v\|_{f,p,\alpha}^2. \quad (8.1.6)$$

However, third covariant derivatives are *not* symmetric when M is a curved manifold, as follows from the Ricci identity [Lee18, Thm. 7.14]. To see this, we combine Eqs. (6.4.1) and (8.1.4) to see that for any three vector fields X, Y, Z :

$$\begin{aligned} (\nabla^3 f)(X, Y, Z) &= X(Y(Zf)) - X((\nabla_Y Z)f) - (\nabla_X Y)(Zf) + (\nabla_{\nabla_X Y} Z)f \\ &\quad - Y((\nabla_X Z)f) + (\nabla_Y (\nabla_X Z))f. \end{aligned}$$

Using symmetry of the Levi-Civita connection, one finds that

$$(\nabla^3 f)(X, Y, Z) - (\nabla^3 f)(Y, X, Z) = -(\mathbf{R}(X, Y)Z)f = -\langle \mathbf{R}(X, Y)Z, \text{grad}(f) \rangle \quad (8.1.7)$$

Accordingly, the third covariant derivative is in general *not* symmetric. Indeed, the asymmetry is precisely related to the nonvanishing of the Riemann curvature tensor!

Due to this asymmetry, to establish self-concordance, we have to show Eq. (8.1.5) for possibly different $u, v \in T_p M$, whereas we could assume $u = v$ in the Euclidean case; see Section 8.1.2 for more details. The following proof of Lemma 8.1.2(ii) is a generalization of [Nes18, Thm. 5.1.1] to our setting.

Proof of Lemma 8.1.2(ii). For $p \in D = D_1 \cap D_2$ and $u, v \in T_p M$, we have

$$\begin{aligned} & \frac{|(\nabla^3 f)_p(u, v, v)|}{2\sqrt{(\nabla^2 f)_p(u, u)(\nabla^2 f)_p(v, v)}} \\ & \leq \frac{|(\nabla^3 f_1)_p(u, v, v)| + |(\nabla^3 f_2)_p(u, v, v)|}{2\sqrt{(\nabla^2 f_1)_p(u, u) + (\nabla^2 f_2)_p(u, u)((\nabla^2 f_1)_p(v, v) + (\nabla^2 f_2)_p(v, v))}} \\ & \leq \frac{x_1 \omega_1 / \sqrt{\alpha_1} + x_2 \omega_2 / \sqrt{\alpha_2}}{\sqrt{x_1^2 + x_2^2(\omega_1 + \omega_2)}}, \end{aligned} \quad (8.1.8)$$

where we let $x_i := \sqrt{(\nabla^2 f_i)_p(u, u)}$ and $\omega_i := (\nabla^2 f_i)_p(v, v)$ for $i = 1, 2$, and for the last estimate we used α_i -self-concordance of f_i . We now upper bound the quantity in Eq. (8.1.8). Observing invariance under the change $(x_1, x_2, \omega_1, \omega_2) \rightarrow (sx_1, sx_2, t\omega_1, t\omega_2)$ for $s, t > 0$, we may consider the following optimization problem:

$$\begin{aligned} & \text{maximize} && \omega_1 x_1 / \sqrt{\alpha_1} + \omega_2 x_2 / \sqrt{\alpha_2} \\ & \text{s.t.} && x_1^2 + x_2^2 = 1, \quad \omega_1 + \omega_2 = 1, \\ & && x_1, x_2, \omega_1, \omega_2 \geq 0. \end{aligned}$$

First we fix ω_i , and maximize over the choice of x_i . This is a linear maximization over the intersection of the unit circle with the positive orthant, with objective given by $(\omega_1 / \sqrt{\alpha_1}, \omega_2 / \sqrt{\alpha_2})$, which is itself in the positive orthant. Therefore the maximum is attained at

$$(x_1, x_2) = \frac{(\omega_1 / \sqrt{\alpha_1}, \omega_2 / \sqrt{\alpha_2})}{\sqrt{\omega_1^2 / \alpha_1 + \omega_2^2 / \alpha_2}},$$

where the value of the objective is $\sqrt{\omega_1^2 / \alpha_1 + \omega_2^2 / \alpha_2}$. This reduces the problem to

$$\text{maximize } \sqrt{\omega_1^2 / \alpha_1 + \omega_2^2 / \alpha_2} \quad \text{s.t. } \omega_1 + \omega_2 = 1, \quad \omega_1, \omega_2 \geq 0.$$

By convexity of the objective, the maximum is attained at $(\omega_1, \omega_2) = (1, 0)$ or $(\omega_1, \omega_2) = (0, 1)$. Therefore Eq. (8.1.8) is at most $\max(1/\sqrt{\alpha_1}, 1/\sqrt{\alpha_2})$, and f is α -self-concordant for $\alpha = \min(\alpha_1, \alpha_2)$. The claim that f is strongly α -self-concordant whenever the f_i are strongly α_i -self-concordant then follows from Lemma 6.5.2. \square

We now state a key property that is required for the analysis of Newton's method of self-concordant functions. It quantifies the change of the Hessian or local norm as a function of the distance, measured with respect to the norm (8.1.2), providing a finitary version of Definition 8.1.1, generalizing an important property (Definition 4.2.1) from the Euclidean setting. Then the following result is a direct translation of the Euclidean argument in [NN94, Thm. 2.1.1] along with the notion of self-concordance from Definition 8.1.1.

Theorem 8.1.4 (Stability of Hessians). *Let $f: D \rightarrow \mathbb{R}$ be an α -self-concordant function defined on an open convex subset $D \subseteq M$, and let $p \in D$. Let $u \in T_p M$ be such that $r := \|u\|_{f,p,\alpha} < 1$. If $q := \text{Exp}_p(u) \in D$, then we have the following estimate: for all $v \in T_p M$,*

$$(1-r)^2 (\nabla^2 f)_p(v, v) \leq (\nabla^2 f)_q(\tau_{\gamma,1} v, \tau_{\gamma,1} v) \leq \frac{1}{(1-r)^2} (\nabla^2 f)_p(v, v), \quad (8.1.9)$$

or, equivalently,

$$(1-r)^2 (\nabla^2 f)_p \leq \tau_{\gamma,1}^* (\nabla^2 f)_q \leq \frac{1}{(1-r)^2} (\nabla^2 f)_p,$$

where $\tau_{\gamma,1}$ denotes the parallel transport along the geodesic $\gamma(t) := \text{Exp}_p(tu)$ from p to q .

Proof. Since the domain is convex, we know that $\gamma(t) = \text{Exp}_p(tu) \in D$ for all $t \in [0, 1]$. Consider the following two functions:

$$\begin{aligned} \phi: [0, 1] &\rightarrow \mathbb{R}, & \phi(t) &= (\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t} v, \tau_{\gamma,t} v), \\ \psi: [0, 1] &\rightarrow \mathbb{R}, & \psi(t) &= (\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t} u, \tau_{\gamma,t} u). \end{aligned}$$

Using Eq. (6.3.1), with $T = \nabla^2 f$ and using that $\dot{\gamma}(t) = \tau_{\gamma,t} u$, we have

$$\dot{\phi}(t) = \left(\nabla_{\dot{\gamma}(t)} (\nabla^2 f) \right) (\tau_{\gamma,t} v, \tau_{\gamma,t} v) = (\nabla^3 f)(\tau_{\gamma,t} u, \tau_{\gamma,t} v, \tau_{\gamma,t} v).$$

Hence, using α -self-concordance as in Eq. (8.1.1),

$$|\dot{\phi}(t)| \leq \frac{2}{\sqrt{\alpha}} \sqrt{\psi(t)} \phi(t). \quad (8.1.10)$$

Similarly,

$$\dot{\psi}(t) = \left(\nabla_{\dot{\gamma}(t)} (\nabla^2 f) \right) (\tau_{\gamma,t} u, \tau_{\gamma,t} u) = (\nabla^3 f)(\tau_{\gamma,t} u, \tau_{\gamma,t} u, \tau_{\gamma,t} u).$$

and hence using only α -self-concordance along the geodesic γ , as in Eq. (8.1.13), we find that

$$|\dot{\psi}(t)| \leq \frac{2}{\sqrt{\alpha}} \psi(t)^{3/2}. \quad (8.1.11)$$

With these estimates in place we can proceed as in the proof of [NN94, Thm. 2.1.1]. By Grönwall's inequality, there are two cases: either ψ vanishes identically on the interval $[0, 1]$, or it is everywhere positive. In the former case, Eq. (8.1.10) implies that ϕ is constant and hence $\phi(1) = \phi(0)$, which in turn implies the claim. In the latter case, we can write Eq. (8.1.11) as

$$\left| \partial_t \psi(t)^{-1/2} \right| = \frac{1}{2} \frac{|\dot{\psi}(t)|}{\psi(t)^{3/2}} \leq \frac{1}{\sqrt{\alpha}}, \quad (8.1.12)$$

from which it follows that

$$\psi(t)^{-1/2} \geq \psi(0)^{-1/2} - \frac{t}{\sqrt{\alpha}} = \frac{1}{\sqrt{\alpha} \|u\|_{f,p,\alpha}} - \frac{t}{\sqrt{\alpha}} = \frac{1-rt}{r\sqrt{\alpha}}$$

and hence, since $r < 1$,

$$\sqrt{\psi(t)} \leq \frac{r\sqrt{\alpha}}{1-rt}.$$

Thus Eq. (8.1.10) implies

$$|\dot{\phi}(t)| \leq \frac{2r}{1-rt} \phi(t).$$

Similarly to the above, either ϕ vanishes identically on $[0, 1]$, in which case there is nothing to prove, or it is everywhere positive, in which case we have

$$\left| \partial_t \log \phi(t) \right| \leq \frac{2r}{1-rt}$$

and hence

$$\left| \log \frac{\phi(t)}{\phi(0)} \right| \leq 2 \log \frac{1}{1-rt}.$$

For $t = 1$ this yields the desired inequality. \square

8.1.2. Self-concordance along geodesics

When $M = \mathbb{R}^n$ is a Euclidean space, then the third derivative is symmetric in all three arguments, and standard results on trilinear forms [Ban38] imply that the above is equivalent to $|\partial_{t=0}^3 f(p + tv)| = |(\nabla^3 f)_p(v, v, v)| \leq 2\alpha \|v\|_{f,p,\alpha}^3$ for all $p, v \in \mathbb{R}^n$, which shows that self-concordance is equivalent to *self-concordance along the geodesics* of Euclidean space. This characterization is highly useful for showing that functions are self-concordant. The richness of the family of self-concordant functions is a key reason for the wide applicability of interior-point methods [NN94; Hil14; Fox15; BE19; Che23].

This notion can also be generalized naturally to the Riemannian setting:

Definition 8.1.5 (Self-concordance along geodesics). Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex subset $D \subseteq M$, and let $\alpha > 0$. We say that f is α -self-concordant along geodesics if, for all $p \in D$ and for all $v \in T_p M$, we have

$$\left| \partial_{t=0}^3 f(\text{Exp}_p(tv)) \right| = |(\nabla^3 f)_p(v, v, v)| \leq \frac{2}{\sqrt{\alpha}} \left((\nabla^2 f)_p(v, v) \right)^{3/2} \quad (8.1.13)$$

or, equivalently,

$$\left| \partial_{t=0}^3 f(\text{Exp}_p(tv)) \right| = |(\nabla^3 f)_p(v, v, v)| \leq 2\alpha \|v\|_{f,p,\alpha}^3. \quad (8.1.14)$$

It is called *strongly α -self-concordant along geodesics* if is not just convex but closed convex, that is, if its epigraph (6.5.1) is a closed subset of $M \times \mathbb{R}$.

In other words, f is (strongly) α -self-concordant along geodesics if and only if for every geodesic $\gamma: \mathbb{R} \rightarrow M$, the function $f \circ \gamma: \mathbb{R} \rightarrow \mathbb{R}$ is (strongly) α -self-concordant on $I := \gamma^{-1}(D)$. There is also a version of Lemma 8.1.2 as a direct consequence of the Euclidean result.

Definition 8.1.5 had been proposed in [Ji07; JMJ07] as a suitable notion of self-concordance in the Riemannian setting. Clearly, any (strongly) self-concordant function is also (strongly) self-concordant along geodesics. However, since third covariant derivatives are *not* symmetric in all arguments when M is a curved manifold, as we saw in Eq. (8.1.7), self-concordance along geodesics need *not* imply self-concordance in the stronger sense of Definition 8.1.1, in contrast to what was suggested in [JMJ07, Eq. (3) and Prop. 1]. While self-concordance along geodesics already allows lifting several useful results from the Euclidean theory, it is the stronger notion of Definition 8.1.1 that is required to prove the fundamental Theorem 8.1.4, which underpins the analysis of the Newton method in the quadratic convergence regime in Theorem 8.1.16. We give non-trivial examples of self-concordant functions on curved spaces in Chapters 9 and 10.

In the remainder of this subsection we discuss a number of useful results for functions that are self-concordant along geodesics. These follow directly from the Euclidean theory. While some of these were already proved in [Ji07; JMJ07], we give all proofs to keep the exposition self-contained. We start with a version of [Nes18, Thm. 5.1.5].

Proposition 8.1.6 (Stability of second derivative along geodesic). *Let $f: D \rightarrow \mathbb{R}$ be α -self-concordant along geodesics, with $D \subseteq M$ open and convex, and let $p \in D$. Consider any geodesic $\gamma(t) = \text{Exp}_p(tu)$ such that $\gamma(1) \in D$, and set $r := \|u\|_{f,p,\alpha}$. Then the α -self-concordant function $g(t) := f(\gamma(t))$ for $t \in [0, 1]$ satisfies the lower bound*

$$\ddot{g}(t) \geq \frac{\ddot{g}(0)}{(1+tr)^2} = \frac{\alpha r^2}{(1+tr)^2}, \quad (8.1.15)$$

and if $rt < 1$ also the upper bound

$$\ddot{g}(t) \leq \frac{\ddot{g}(0)}{(1-tr)^2} = \frac{\alpha r^2}{(1-tr)^2}. \quad (8.1.16)$$

Proof. As in the proof of Theorem 8.1.4, we consider the function

$$\psi: [0, 1] \rightarrow \mathbb{R}, \quad \psi(t) = \ddot{g}(t),$$

and find from Eq. (8.1.11) that it either vanishes identically on $[0, 1]$, in which case the claim holds trivially, or it is everywhere positive, in which case Eq. (8.1.12) holds, namely for all $t \in [0, 1]$,

$$\left| \partial_t \psi(t)^{-1/2} \right| \leq \frac{1}{\sqrt{\alpha}}.$$

Accordingly,

$$\psi(0)^{-1/2}(1 - \text{tr}) = \psi(0)^{-1/2} - \frac{t}{\sqrt{\alpha}} \leq \psi(t)^{-1/2} \leq \psi(0)^{-1/2} + \frac{t}{\sqrt{\alpha}} = \psi(0)^{-1/2}(1 + \text{tr}),$$

which implies both bounds. \square

The lower bound strengthens the one in Eq. (8.1.9) in the special case that $v = u$. The upper bound implies that any function that is strongly self-concordant along geodesics must contain a certain region in its domain. We first define the region and then state the result.

Definition 8.1.7 (Dikin ellipsoid). Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex subset $D \subseteq M$, and let $\alpha > 0$. Then the (open) Dikin ellipsoid of radius $r > 0$ at $p \in M$ is

$$B_{f,p,\alpha}^\circ(r) = \left\{ \text{Exp}_p(u) : u \in T_p M, \|u\|_{f,p,\alpha} < r \right\}.$$

For $\alpha = 1$, we abbreviate $B_{f,p}^\circ := B_{f,p,1}^\circ$.

The following result is easily generalized from the Euclidean setting. The proof is essentially the same as in [NN94, Thm. 2.1.1].

Corollary 8.1.8 (Dikin inclusion). Let $f: D \rightarrow \mathbb{R}$ be strongly α -self-concordant along geodesics, defined on an open convex subset $D \subseteq M$. Then $B_{f,p,\alpha}^\circ(1) \subseteq D$ for every $p \in D$.

Proof. Take any $v \in T_p M$ such that $r := \|v\|_{f,p,\alpha} < 1$. Let σ be the supremum of those $s \geq 0$ such that $\gamma(s) := \text{Exp}_p(sv) \in D$. Since $p \in D$ and D is open, we know that $\sigma > 0$, and since D is convex, we know that $\gamma(s) \in D$ for all $s \in [0, \sigma)$.

We need to show that $\gamma(1) \in D$ and claim that in fact $\sigma > 1/r > 1$ (with $1/0 = \infty$). For sake of finding a contradiction, assume that this is not so, i.e., that $\sigma \leq 1/r$. For every $s \in [0, \sigma)$ we can apply Proposition 8.1.6 with $u := sv$, which satisfies $\|u\|_{f,p,\alpha} = sr < \sigma r \leq 1$. Then the upper bound in Eq. (8.1.16) gives

$$\ddot{g}(s) \leq \frac{1}{(1 - sr)^2} \ddot{g}(0),$$

where $g(s) = f(\gamma(s))$. Accordingly, the function g has bounded derivative on $[0, \sigma)$, thus it is itself bounded on this interval, say $g(s) \leq L$ for some $L \in \mathbb{R}$. As f is strongly self-concordant, the level set $\{q \in D : f(q) \leq L\}$ is closed in M , and hence it must contain $\gamma(\sigma) = \lim_{s \uparrow \sigma} \gamma(s)$. But D is open, so this in turn implies there must also exist some $t > \sigma$ such that $\gamma(t) \in D$, contradicting the definition of σ . \square

In other words, for any $p \in D$ and $u \in T_p M$ such that $\|u\|_{f,p,\alpha} < 1$ it is automatically true that $\text{Exp}_p(u) \in D$, so we do not have to assume this in Theorem 8.1.4 and Proposition 8.1.6.

The above also implies that a strongly-self-concordant function can only have a degenerate Hessian if its domain contains a geodesic.

Corollary 8.1.9 (Domain). If a strongly α -self-concordant function $f: D \rightarrow \mathbb{R}$ contains no (infinite) geodesic in its domain, then $(\nabla^2 f)_p$ is positive definite for all $p \in D$. In particular, this is the case if M is a Hadamard manifold and the domain is bounded.

8. Interior-point methods on manifolds: the framework

Proof. If $(\nabla^2 f)_p(u, u) = 0$ for some $p \in D$ and $u \in T_p M$, then $\text{Exp}_p(\mathbb{R}u) \subseteq B_{f,p,\alpha}^\circ(1)$. Thus Corollary 8.1.8 shows that D contains the geodesic $\gamma(t) = \text{Exp}_p(tu)$ for $t \in \mathbb{R}$. \square

The following results bound a self-concordant function in terms of its linear approximation at some arbitrary point, in terms of the quantity

$$\rho: (-\infty, 1) \rightarrow \mathbb{R}, \quad \rho(r) = -r - \log(1 - r), \quad (8.1.17)$$

which is $\rho(r) = \frac{1}{2}r^2 + O(r^3)$ for small r . The first result lifts [Nes18, Thm. 5.1.8] to the geodesic setting and follows directly by integrating the lower bound in Proposition 8.1.6.

Corollary 8.1.10 (Lower bound). *Let $f: D \rightarrow \mathbb{R}$ be α -self-concordant along geodesics, defined on an open convex subset $D \subseteq M$, and let $p \in D$. Then, for every $u \in T_p M$ such that $q := \text{Exp}_p(u) \in D$, we have*

$$df_q(\tau_{\gamma,t}u) - df_p(u) \geq \frac{\alpha tr^2}{1 + tr} \quad (8.1.18)$$

where $r := \|u\|_{f,p,\alpha}$ and $\tau_{\gamma,t}$ denotes the parallel transport along the geodesic $\gamma(t) := \text{Exp}_p(tu)$ from p to q , and

$$f(q) \geq f(p) + df_p(u) + \alpha\rho(-r).$$

Proof. By Proposition 8.1.6, we see that $g(t) := f(\text{Exp}_p(tu))$ satisfies

$$\ddot{g}(t) \geq \frac{\alpha r^2}{(1 + tr)^2}$$

for all $t \in [0, 1]$. By integrating,

$$\dot{g}(t) - \dot{g}(0) \geq \int_0^t \frac{\alpha r^2}{(1 + sr)^2} ds = \frac{\alpha tr^2}{1 + tr}.$$

Since $\dot{g}(0) = df_p(u)$ and $\dot{g}(1) = df_q(\tau_{\gamma,1}u)$, this proves the first bound. One more integral yields

$$g(1) - g(0) - \dot{g}(0) \geq \int_0^1 \frac{\alpha sr^2}{1 + sr} ds = \alpha(r - \log(1 + r)) = \alpha\rho(-r). \quad \square$$

The second result generalizes [Nes18, Thm. 5.1.9] to the geodesic setting and follows by similarly integrating the upper bound in Proposition 8.1.6.

Corollary 8.1.11 (Upper bound). *Let $f: D \rightarrow \mathbb{R}$ be α -self-concordant along geodesics, defined on an open convex subset $D \subseteq M$, and let $p \in D$. Then, for every $u \in T_p M$ such that $q := \text{Exp}_p(u) \in D$ and $r := \|u\|_{f,p,\alpha} < 1$, we have*

$$df_q(\tau_{\gamma,t}u) - df_p(u) \leq \frac{\alpha tr^2}{1 - rt},$$

where $\tau_{\gamma,t}$ denotes the parallel transport along the geodesic $\gamma(t) = \text{Exp}_p(tu)$ from p to q , and

$$f(q) \leq f(p) + df_p(u) + \alpha\rho(r).$$

If f is strongly α -self-concordant along geodesics, then the requirement that $q \in D$ is automatic (by Corollary 8.1.8).

Proof. Similarly to the proof of Corollary 8.1.10, we can apply Proposition 8.1.6 to see that the function $g(t) := f(\text{Exp}_p(tu))$ satisfies

$$\ddot{g}(t) \leq \frac{\alpha r^2}{(1 - tr)^2}$$

for all $t \in [0, 1]$. By integration,

$$\dot{g}(t) - \dot{g}(0) \leq \int_0^t \frac{\alpha r^2}{(1 - sr)^2} ds = \frac{\alpha tr^2}{1 - tr}$$

and

$$g(1) - g(0) - \dot{g}(0) \leq \int_0^1 \frac{\alpha sr^2}{1 - sr} ds = \alpha(-r - \log(1 - r)) = \alpha\rho(r). \quad \square$$

8.1.3. Newton's method

We are now ready to give an analysis of Newton's method for self-concordant functions. In particular, as in the Euclidean case, we are able to provide quadratic guarantees on the changes in the so-called Newton decrement (Theorem 8.1.16). This key result requires self-concordance. Afterwards we also recall some useful results due to [Ji07; JMJ07] which only rely on self-concordance along geodesics.

Recall Newton's method (cf. [Udr94, §7.5]): given a convex function f and a point p in its domain, consider its local quadratic approximation

$$f(\text{Exp}_p(v)) \approx f(p) + df_p(v) + \frac{1}{2}(\nabla^2 f)_p(v, v)$$

and minimize the right-hand side over all $v \in T_p M$. If $(\nabla^2 f)_p$ is non-degenerate and hence positive definite, as we will assume for convenience, there is a unique minimizer called the Newton step.

Definition 8.1.12 (Newton step and Newton iterate). Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex set $D \subseteq M$, and let $p \in D$ be a point such that $(\nabla^2 f)_p$ is positive definite. Then we define the *Newton step* of f at p as the unique vector $n_{f,p} \in T_p M$ such that

$$(\nabla^2 f)_p(n_{f,p}, \cdot) = -df_p \quad (8.1.19)$$

and the *Newton iterate* of f at p is defined as

$$p_{f,+} := \text{Exp}_p(n_{f,p}) \in M,$$

which need not be in D . We can also write

$$n_{f,p} = -\text{Hess}(f)_p^{-1} \text{grad}(f)_p \quad \text{and} \quad p_{f,+} = \text{Exp}_p(-\text{Hess}(f)_p^{-1} \text{grad}(f)_p).$$

in terms of the gradient vector and Hessian operator (see Section 6.4).

The gap between the function value and the minimum of the quadratic approximation is

$$\frac{1}{2}(\nabla^2 f)_p(n_{f,p}, n_{f,p}) = \frac{\alpha}{2} \|n_{f,p}\|_{f,p,\alpha}^2 = \frac{\alpha}{2} \lambda_{f,\alpha}(p)^2,$$

where $\lambda_{f,\alpha}$ is the so-called Newton decrement, which we define next.

Definition 8.1.13 (Newton decrement). Let $f: D \rightarrow \mathbb{R}$ be a convex function defined on an open convex set $D \subseteq M$, let $p \in D$ be a point such that $(\nabla^2 f)_p$ is positive definite, and let $\alpha > 0$. Then we define the *Newton decrement* of f at p by

$$\begin{aligned} \lambda_{f,\alpha}(p) &:= \|\mathfrak{n}_{f,p}\|_{f,p,\alpha} \\ &= \frac{1}{\alpha} \|\mathrm{d}f_p\|_{f,p,\alpha}^* = \max_{0 \neq v \in T_p M} \frac{|\mathrm{d}f_p(v)|}{\alpha \|v\|_{f,p,\alpha}} = \max_{0 \neq v \in T_p M} \frac{|\mathrm{d}f_p(v)|}{\sqrt{\alpha (\nabla^2 f)_p(v,v)}}, \end{aligned}$$

where $\|\omega\|_{f,p,\alpha}^* := \max_{0 \neq v \in T_p M} \frac{|\omega(v)|}{\|v\|_{f,p,\alpha}}$ is the dual norm of $\omega \in T_p^* M$ induced by $\|\cdot\|_{f,p,\alpha}$. That is,¹

$$\lambda_{f,\alpha}(p) = \min\{\lambda \geq 0 : \mathrm{d}f_p \otimes \mathrm{d}f_p \leq \lambda^2 \alpha (\nabla^2 f)_p\} \quad (8.1.20)$$

$$= \min\{\lambda \geq 0 : -\mathrm{d}f_p(u) - \frac{1}{2}(\nabla^2 f)_p(u,u) \leq \frac{\lambda^2 \alpha}{2} \forall u \in T_p M\}. \quad (8.1.21)$$

For $\alpha = 1$, we abbreviate $\lambda_f := \lambda_{f,1}$ and $\|\cdot\|_{f,p}^* := \|\cdot\|_{f,p,1}^*$.

The Newton decrement is invariant under rescaling f in the sense that $\lambda_{f,\alpha} = \lambda_{cf,c\alpha}$ for any constant $c > 0$ (cf. Lemma 8.1.2). When $(\nabla^2 f)_p$ is degenerate, the Newton decrement can still be defined as $\lambda_{f,\alpha}(p) = \inf\{c \geq 0 : |\mathrm{d}f_p(v)| \leq \alpha c \|v\|_{f,p,\alpha} \forall v \in T_p M\}$, which has the same interpretation as explained above; but we will mostly not need this.

Just like in the Euclidean case the Newton decrement provides a certificate for the existence of minimizers and the function gap. This essentially follows from the Euclidean argument [Nes18, Thm. 5.1.13].

Proposition 8.1.14 (Existence of minimizers). *Let $f: D \rightarrow \mathbb{R}$ be α -self-concordant along geodesics, defined on an open convex subset $D \subseteq M$. If $p \in D$ is such that $\lambda_{f,\alpha}(p) < 1$, then f is bounded from below: we have*

$$f_* := \inf_{q \in D} f(q) \geq f(p) - \alpha \rho(\lambda_{f,\alpha}(p)), \quad (8.1.22)$$

where ρ is the quantity defined in Eq. (8.1.17). If in addition f is strongly α -self-concordant along geodesics and $(\nabla^2 f)_p$ is positive definite, then the function attains its minimum at some $p_* \in D$.

Proof. We abbreviate $\lambda := \lambda_{f,\alpha}(p)$ and $r := \|u\|_{f,p,\alpha}$. For every $q = \mathrm{Exp}_p(u) \in D$, we have using Corollary 8.1.10 and the definition of the Newton decrement the lower bound

$$f(q) - f(p) \geq \mathrm{d}f_p(u) + \alpha \rho(-r) \geq -\alpha r \lambda + \alpha \rho(-r) = \alpha \delta(r), \quad (8.1.23)$$

where

$$\delta(r) = r(1 - \lambda) - \log(1 + r).$$

If $\lambda < 1$, $\delta(r)$ is minimized at $r = \lambda/(1 - \lambda)$, and we obtain

$$f(q) - f(p) \geq \alpha(\lambda + \log(1 - \lambda)) = -\alpha \rho(\lambda).$$

¹To see the second equality, replace u by tu for $t \in \mathbb{R}$, and maximize over t .

This implies Eq. (8.1.22).

On the other hand, $\delta(r) \rightarrow \infty$ as $r \rightarrow \infty$, so Eq. (8.1.23) shows that the level set $\{q \in D : f(q) \leq f(p)\}$ is contained in a Dikin ellipsoid of some suitable radius. If we assume that $(\nabla^2 f)_p$ is positive definite then Dikin ellipsoids are bounded. Thus if f is also α -strongly self-concordant along geodesics then Lemma 6.5.1 (iii) shows that f attains its minimum at some $p_* \in D$. \square

The minimizer in Proposition 8.1.14 is unique assuming strict convexity, as follows, e.g., if $\nabla^2 f$ is positive definite throughout the domain. The Newton decrement also certifies closeness to minimizers if they exist:

Lemma 8.1.15. *Let $f: D \rightarrow \mathbb{R}$ be α -self-concordant along geodesics, defined on an open convex subset $D \subseteq M$, and let $p \in D$ be such that $\lambda_{f,\alpha}(p) < 1$. If f attains a minimum at $p_* = \text{Exp}_p(u)$ for $u \in T_p M$, then*

$$\|u\|_{f,p,\alpha} \leq \frac{\lambda_{f,\alpha}(p)}{1 - \lambda_{f,\alpha}(p)}.$$

Proof. Consider the geodesic $\gamma(t) = \text{Exp}_p(tu)$ from p to p_* . Then by Corollary 8.1.10, we have

$$\frac{\alpha r^2}{1+r} \leq df_{p_*}(\tau_{\gamma,1}u) - df_p(u) = -df_p(u) \leq |df_p(u)| \leq \alpha r \lambda_{f,\alpha}(p),$$

where $r := \|u\|_{f,p,\alpha}$; the equality follows because $df_{p_*} = 0$ because p_* is a minimizer of f . Thus we have

$$\frac{r}{1+r} \leq \lambda_{f,\alpha}(p)$$

and for $\lambda_{f,\alpha}(p) < 1$ this implies the desired bound. \square

The following theorem is key to the analysis of Newton's method for self-concordant functions. It bounds the Newton decrement after one Newton step quadratically in terms of the original Newton decrement. This requires self-concordance in the sense of Definition 8.1.1, rather than the weaker notion along geodesics, as its proof involves comparing the length of the new Newton step transported along the geodesics given by the previous Newton step, i.e., there are two natural directions. The proof adapts the Euclidean argument in [Ren01, Thm. 2.2.4].

Theorem 8.1.16. *Let $f: D \rightarrow \mathbb{R}$ be a strongly α -self-concordant function defined on an open convex set $D \subseteq M$, with positive definite Hessian. Let $p \in D$ be a point such that $\lambda_{f,\alpha}(p) < 1$. Then the Newton iterate remains in the domain, i.e., $p_{f,+} \in D$, and moreover*

$$\lambda_{f,\alpha}(p_{f,+}) \leq \left(\frac{\lambda_{f,\alpha}(p)}{1 - \lambda_{f,\alpha}(p)} \right)^2.$$

Proof. We abbreviate the Newton step, iterate, and increment by $n_p := n_{f,p}$, $p_+ := p_{f,+}$, and $\lambda := \lambda_{f,\alpha}(p)$, respectively. Corollary 8.1.8 along with the definitions shows that $p_+ \in D$. Then the entire geodesic segment $\gamma(t) := \text{Exp}_p(tn_p)$ for $t \in [0, 1]$ is

contained in the domain D . We now prove the desired estimate, starting with Theorem 8.1.4, which gives the upper bound

$$\begin{aligned} \lambda_{f,\alpha}(p_+) &= \max_{w \in T_{p_+}M} \frac{|df_{p_+}(w)|}{\alpha \|w\|_{f,p_+,\alpha}} = \max_{v \in T_pM} \frac{|df_{p_+}(\tau_{\gamma,1}v)|}{\alpha \|\tau_{\gamma,1}v\|_{f,p_+,\alpha}} \\ &\leq \frac{1}{1-\lambda} \max_{v \in T_pM} \frac{|df_{p_+}(\tau_{\gamma,1}v)|}{\alpha \|v\|_{f,p,\alpha}}, \end{aligned} \quad (8.1.24)$$

where $\tau_{\gamma,1}$ denotes parallel transport along the geodesic γ from p to p_+ . Next, we observe that by the fundamental theorem of calculus, Eq. (6.3.1), and Eq. (8.1.19), for all $v \in T_pM$,

$$\begin{aligned} df_{p_+}(\tau_{\gamma,1}v) &= df_{p_+}(\tau_{\gamma,1}v) - df_p(v) + df_p(v) \\ &= \int_0^1 \partial_t df_{\gamma(t)}(\tau_{\gamma,t}v) dt + df_p(v) \\ &= \int_0^1 (\nabla_{\dot{\gamma}(t)} df)_{\gamma(t)}(\tau_{\gamma,t}v) dt + df_p(v) \\ &= \int_0^1 (\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t}n_p, \tau_{\gamma,t}v) dt + df_p(v) \\ &= \int_0^1 [(\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t}n_p, \tau_{\gamma,t}v) - (\nabla^2 f)_p(n_p, v)] dt \\ &= \beta(n_p, v), \end{aligned} \quad (8.1.25)$$

where we have introduced the symmetric bilinear form

$$\beta: T_pM \times T_pM \rightarrow \mathbb{R}, \quad \beta(u, v) = \int_0^1 [(\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t}u, \tau_{\gamma,t}v) - (\nabla^2 f)_p(u, v)] dt.$$

By Theorem 8.1.4 and using $\|tn_p\|_{f,p,\alpha} = t\lambda$, we have, for all $v \in T_pM$,

$$\begin{aligned} [(1-t\lambda)^2 - 1](\nabla^2 f)_p(v, v) &\leq (\nabla^2 f)_{\gamma(t)}(\tau_{\gamma,t}v, \tau_{\gamma,t}v) - (\nabla^2 f)_p(v, v) \\ &\leq \left[\frac{1}{(1-t\lambda)^2} - 1 \right] (\nabla^2 f)_p(v, v). \end{aligned}$$

By integrating the lower and upper bounds from $t = 0$ to $t = 1$,

$$-\left(\lambda - \frac{\lambda^2}{3}\right) (\nabla^2 f)_p(v, v) \leq \beta(v, v) \leq \left(\frac{\lambda}{1-\lambda}\right) (\nabla^2 f)_p(v, v).$$

One may verify that $\max\{\lambda - \lambda^2/3, \lambda/(1-\lambda)\} = \lambda/(1-\lambda)$ as $\lambda < 1$. Together with the Cauchy-Schwarz inequality, this implies that for all $u, v \in T_pM$,

$$|\beta(u, v)| \leq \frac{\lambda}{1-\lambda} \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(v, v)} = \frac{\alpha\lambda}{1-\lambda} \|u\|_{f,p,\alpha} \|v\|_{f,p,\alpha}.$$

Together with Eqs. (8.1.24) and (8.1.25), we obtain the upper bound

$$\lambda_{f,\alpha}(p_+) \leq \frac{1}{1-\lambda} \max_{v \in T_pM} \frac{|\beta(n_p, v)|}{\alpha \|v\|_{f,p,\alpha}} \leq \frac{\lambda}{(1-\lambda)^2} \|n_p\|_{f,p,\alpha} = \frac{\lambda^2}{(1-\lambda)^2}. \quad \square$$

Theorem 8.1.16 implies that the Newton method converges quadratically for sufficiently small λ . For example, suppose that $\lambda \leq \lambda_* := 1 - \frac{1}{\sqrt{2}}$. Then we have

$$\left(\frac{\lambda}{1-\lambda}\right)^2 \leq \left(\frac{\lambda}{1-\lambda_*}\right)^2 = 2\lambda^2 \leq \lambda_*, \quad (8.1.26)$$

meaning the Newton decrement decreases quadratically and stays below λ_* , so we can iterate. This implies the following result (cf. [NN94, Thm. 2.2.3]):

Theorem 8.1.17 (Quadratic convergence of the Newton method). *Let $f: D \rightarrow \mathbb{R}$ be a strongly α -self-concordant function defined on an open convex set $D \subseteq M$, with positive definite Hessian. Let $p_0 \in D$ be a point such that $\lambda_{f,\alpha}(p_0) \leq \lambda_* := 1 - 1/\sqrt{2} \approx 0.293$. Then the Newton iterations*

$$p_{t+1} := \text{Exp}_{p_t}(n_{f,p_t})$$

are well-defined for all $t \in \mathbb{N}$ (i.e., each $p_t \in D$) and we have

$$\lambda_{f,\alpha}(p_t) \leq \frac{1}{2}(2\lambda_{f,\alpha}(p_0))^{2^t} \leq \frac{1}{2}(2\lambda_*)^{2^t}.$$

In particular, $O(\log \log \frac{\alpha}{\varepsilon})$ Newton iterations suffice to find a point p_t such that $f(p_t) \leq f_* + \varepsilon$, for $\varepsilon < \alpha/e$.

Proof. We abbreviate $\lambda_t := \lambda_{f,\alpha}(p_t)$. By Theorem 8.1.16 and Eq. (8.1.26), one can see inductively that $p_t \in D$ is well-defined for all $t \in \mathbb{N}$ and that we have $\lambda_t \leq \lambda_*$ and

$$2\lambda_t \leq (2\lambda_{t-1})^2 \leq \dots \leq (2\lambda_0)^{2^t} \leq (2\lambda_*)^{2^t},$$

as claimed. This also implies the last statement, since to achieve $f(p_t) \leq f_* + \varepsilon$ it suffices to have $\rho(\lambda_t) \leq \varepsilon/\alpha$, by Proposition 8.1.14, and we have $\rho(\lambda_t) \leq \lambda_t^2$ for $\lambda_t \leq \lambda_*$. \square

What if we have a starting point such that the Newton decrement does *not* guarantee quadratic convergence? In this case it is well-known that one can employ a *damped* Newton method, with a step size that ensures that one stays inside the Dikin ellipsoid (and hence in the domain) at each step. This works just the same in the Riemannian setting and only requires self-concordance along geodesics (cf. [Nes18, Thm. 5.1.15]):

Theorem 8.1.18 (Damped Newton method). *Let $f: D \rightarrow \mathbb{R}$ be strongly α -self-concordant along geodesics, defined on an open convex set $D \subseteq M$, with positive definite Hessian. Let $p_0 \in D$ be an arbitrary starting point. Then the damped Newton iterations*

$$p_{t+1} := \text{Exp}_{p_t}(u_t) \quad \text{where} \quad u_t := \frac{1}{1 + \lambda_{f,\alpha}(p_t)} n_{f,p_t}$$

are well-defined for all $t \in \mathbb{N}$ (i.e., each $p_t \in D$) and we have

$$f(p_{t+1}) \leq f(p_t) - \alpha\rho(-\lambda_t),$$

where ρ is the quantity defined in Eq. (8.1.17). In particular, if f is bounded from below and we set $f_* := \inf_{p \in D} f(p)$, then $O((f(p_0) - f_*)/\alpha)$ damped Newton iterations suffice to find a point p_t such that $\lambda_{f,\alpha}(p_t) \leq \lambda_*$ (or any other constant).

Proof. We abbreviate $\lambda_t := \lambda_{f,\alpha}(p_t)$. Using Corollary 8.1.11 one can see inductively that $r := \|\mathbf{u}_t\|_{f,p,\alpha} = \lambda_t/(1 + \lambda_t) < 1$ and $p_t \in D$ is well-defined for all $t \in \mathbb{N}$. Moreover,

$$\begin{aligned} f(p_{t+1}) &\leq f(p_t) + df_{p_t}(\mathbf{u}_t) + \alpha\rho(r) \\ &= f(p_t) - (\nabla^2 f)_p(n_{f,p_t}, \mathbf{u}_t) + \alpha\rho(r) \\ &= f(p_t) - \alpha \left(\frac{\lambda_t^2}{1 + \lambda_t} - \rho(r) \right) \\ &= f(p_t) - \alpha(\lambda_t - \log(1 + \lambda_t)) \\ &= f(p_t) - \alpha\rho(-\lambda_t). \end{aligned} \quad \square$$

In particular, Theorem 8.1.18 and Corollary 8.1.8 have the following structural consequence.

Corollary 8.1.19. *Let $f: D \rightarrow \mathbb{R}$ be strongly α -self-concordant along geodesics, defined on an open convex set $D \subseteq M$, with positive definite Hessian. Then f is bounded from below if and only if it attains its minimum (necessarily at a unique minimizer, by strict convexity).*

By combining Theorems 8.1.17 and 8.1.18, we see that we can approximately minimize any strongly α -self-concordant function with positive definite Hessian by first using damped Newton steps from an arbitrary starting point p_0 until we arrive at point with Newton decrement $\leq \lambda_*$; then we are in the quadratic convergence regime and we can take ordinary Newton steps until we arrive at a point p_t with $\rho(\lambda_{f,\alpha}(p_t)) \leq \varepsilon/\alpha$, so that p_t is an ε -approximate minimizer. This requires $O((f(p_0) - f_*)/\alpha + \log \log(\alpha/\varepsilon))$ Newton iterations.

8.2. Barriers, compatibility, and the path-following method

The methods developed in Section 8.1 are sufficient to optimize strongly self-concordant functions. However, it is difficult to guarantee that one starts in the quadratic convergence regime for Newton's method, and the damped Newton method has a worst-case complexity which depends on the gap in function value. Moreover, most convex optimization problems do *not* take the form of a minimization of a strongly self-concordant function over its natural domain. Rather, one is given a convex objective f and a domain D and wants to minimize the former over the latter.

In this section, we show how to circumvent these two issues, assuming one has a *self-concordant barrier* for the domain over which one optimizes. To this end, we generalize the analysis of so-called *path-following (interior-point) methods* [NN94] from the Euclidean to the Riemannian setting. We treat not only the case of geodesically linear objectives, but the more general class of objectives that are *compatible* with the given self-concordant barrier. This will be useful for the applications discussed in Chapter 10. Throughout this section we assume that M is a connected and geodesically complete Riemannian manifold.

8.2.1. Self-concordant barriers

We first define the notion of a self-concordant barrier. The estimates in this section only require the self-concordance to be along geodesics, and we make explicit whenever this is the case. However, the path-following method presented in Section 8.2.3 requires the stronger notion.

Definition 8.2.1 (Barrier). Let $D \subseteq M$ be an open and convex subset, and let $\theta \geq 0$. We say that a function $F: D \rightarrow \mathbb{R}$ is a *non-degenerate strongly self-concordant barrier with parameter θ* , or in short a θ -barrier, if F is a strongly 1-self-concordant function with positive definite Hessian such that $\lambda_F(p) \leq \sqrt{\theta}$ for all $p \in D$, with $\lambda_F = \lambda_{F,1}$ the Newton decrement (Definition 8.1.13). We say that F is a θ -barrier along geodesics if it is only strongly 1-self-concordant along geodesics.

The parameter of a barrier plays an important role in the complexity analysis of the path-following method that we discuss in Section 8.2.3. The following lemma follows readily from the definition:

Lemma 8.2.2. *Let $F_1: D_1 \rightarrow \mathbb{R}$ be a θ_1 -barrier and let $F_2: D_2 \rightarrow \mathbb{R}$ be a θ_2 -barrier. Then $F_1 + F_2$ is a $(\theta_1 + \theta_2)$ -barrier for $D := D_1 \cap D_2$, assuming D is non-empty.*

Next, we prove an important inequality which involves the barrier parameter. To state the result, we define a Riemannian version of the so-called Minkowski function(al) or gauge function. It measures the inverse distance from a point to the boundary of the domain.

Definition 8.2.3 (Minkowski functional). Let $D \subseteq M$ be an open convex subset. For $p \in D$, we define the *Minkowski functional* by

$$\pi_{D,p}: T_p M \rightarrow \mathbb{R}_{\geq 0}, \quad \pi_{D,p}(u) = \inf \left\{ s \geq 0 : \text{Exp}_p \left(\frac{1}{s} u \right) \in D \right\}.$$

This is well-defined since D is open and hence $\pi_{D,p}(u) < \infty$ for every $u \in T_p M$. Note that if $s := \pi_{D,p}(u) = 0$, then the entire infinite geodesic ray $\gamma(t) = \text{Exp}_p(tu)$ is contained in the domain, while if $s > 0$ then $\text{Exp}_p(\frac{1}{s}u)$ is a point in its boundary $\partial D = \overline{D} \setminus D$. Moreover, if $u \in T_p M$ is such that $\text{Exp}_p(u) \in \overline{D}$, then $\pi_p(u) \leq 1$.

Then we have the following result, which can be deduced directly from its Euclidean version [NN94, §2.3.2]. We provide a self-contained proof for convenience.

Proposition 8.2.4. *Let $D \subseteq M$ be open and convex, and let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics. Then one has, for all $p \in D$ and $u \in T_p M$,*

$$dF_p(u) \leq \theta \pi_{D,p}(u).$$

In particular, if $q = \text{Exp}_p(u) \in \overline{D}$ then

$$dF_p(u) \leq \theta.$$

Proof. The second statement follows from the first by the preceding discussion. To prove the first, let $p \in D$ and $u \in T_p M$. If $dF_p(u) \leq 0$ then there is nothing to prove, so we assume that $dF_p(u) > 0$. Define

$$g(t) := F(\text{Exp}_p(tu)).$$

8. Interior-point methods on manifolds: the framework

Then g is well-defined on the interval $I = [0, \pi_{D,p}(u)^{-1})$, where we interpret $0^{-1} = \infty$. By definition of the Newton decrement and recalling that $\ddot{g}(t) > 0$ as F has positive definite Hessian, we have

$$\frac{(\dot{g}(t))^2}{\ddot{g}(t)} \leq \lambda_F^2(p) = \theta.$$

Since we assumed that $\dot{g}(0) = dF_p(u) > 0$, we find that $\theta > 0$, as well as $\dot{g}(t) > 0$ for all $t \in I$, by convexity. Accordingly, we can write the above as

$$\partial_t \left(\frac{1}{\dot{g}(t)} \right) = -\frac{\ddot{g}(t)}{(\dot{g}(t))^2} \leq -\frac{1}{\theta},$$

which implies that

$$\frac{1}{\dot{g}(t)} = \frac{1}{\dot{g}(0)} + \int_0^t \partial_t \left(\frac{1}{\dot{g}(t)} \right) dt \leq \frac{1}{\dot{g}(0)} - \frac{t}{\theta},$$

and hence

$$\dot{g}(t) \geq \frac{1}{\frac{1}{\dot{g}(0)} - \frac{t}{\theta}} = \frac{\theta \dot{g}(0)}{\theta - t \dot{g}(0)}.$$

As the right-hand side diverges as t approaches $\theta/\dot{g}(0)$, we must have $t < \theta/\dot{g}(0)$ for all $t \in I$. Hence

$$\pi_{D,p}(u)^{-1} \leq \frac{\theta}{\dot{g}(0)},$$

which is the desired bound. \square

As a consequence, non-trivial barriers must have positive parameter:

Corollary 8.2.5. *Let $D \subseteq M$ be open and convex, and let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics with $\theta = 0$. Then F is constant and $D = M$.*

Proof. Proposition 8.2.4 shows that $dF = 0$, hence F is locally constant and $\nabla^2 F = 0$. Because F is strongly self-concordant, we may apply Corollary 8.1.8 to conclude that $\text{Exp}_p(T_p M) \subseteq D$ and hence $D = M$, since M is connected and geodesically complete. \square

The minimizer of a barrier, which if it exists is necessarily unique (recall that barriers have positive definite Hessians by definition), plays a special role in the theory.

Definition 8.2.6 (Analytic center). Let $D \subseteq M$ be open and convex, and let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics. If F attains its minimum, then the unique minimizer is called the *analytic center* of D .

Recall that a barrier attains its minimum if and only if it is bounded from below (Corollary 8.1.19). The following result shows that the domain is necessarily enclosed in a Dikin ellipsoid about the analytic center, with radius given by the barrier's parameter. It adapts the Euclidean argument (cf. [Nes18, Thm. 5.3.9], [NN94, Prop. 2.3.2 (iii)]) to the Riemannian setting.

Proposition 8.2.7 (Enclosing Dikin ellipsoid). *Let $D \subseteq M$ be open and convex, and let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics. If $\theta > 0$ and F is bounded from below, with analytic center $p_* \in D$, then*

$$D \subseteq B_{F,p_*}^\circ(2\theta + 1),$$

where $B_{F,p_*}^\circ = B_{F,p_*,1}^\circ$ denotes the Dikin ellipsoid (Definition 8.1.7). That is, the domain is contained in the Dikin ellipsoid with radius $2\theta + 1$ about p_* .

Proof. Let $u \in T_{p_*}M$ be such that $\|u\|_{F,p_*} = 1$, and let $\gamma(t) := \text{Exp}_{p_*}(tu)$. By Corollary 8.1.8, we know that $B_{F,p_*}^\circ(1) \subseteq D$, hence $g(t) := F(\gamma(t))$ is well-defined for $t \in [0, 1)$.

To show that $D \subseteq B_{F,p_*}^\circ(2\theta + 1)$, by convexity of D it suffices to show that $\gamma(1 + 2\theta) \notin D$. From Eq. (8.1.18) in Corollary 8.1.10 and p_* being a minimizer of F , it follows that, for $t \in [0, 1)$,

$$dF_{\gamma(t)}(\tau_{\gamma,t}u) = dF_{\gamma(t)}(\tau_{\gamma,t}u) - dF_{p_*}(u) = \dot{g}(t) - \dot{g}(0) \geq \frac{t}{1+t}.$$

Proposition 8.2.4 on the other hand implies that for

$$dF_{\gamma(t)}(\tau_{\gamma,t}u) \leq \theta \pi_{D,\gamma(t)}(\tau_{\gamma,t}u).$$

Together, we obtain that, for every $t \in [0, 1)$,

$$\theta \pi_{D,\gamma(t)}(\tau_{\gamma,t}u) \geq \frac{t}{1+t}.$$

By the definition of the Minkowski functional, for every $s \in [0, \pi_{D,\gamma(t)}(\tau_{\gamma,t}(u))]$, we have

$$\gamma\left(t + \frac{1}{s}\right) = \text{Exp}_{p_*}\left(\left(t + \frac{1}{s}\right)u\right) = \text{Exp}_{\gamma(t)}\left(\frac{1}{s}\tau_{\gamma,t}u\right) \notin D.$$

Therefore, for every $t \in [0, 1)$ and $s \in [0, \frac{t}{\theta(1+t)})$, we have

$$\gamma\left(t + \frac{1}{s}\right) \notin D.$$

Letting $t \rightarrow 1$ and $s \rightarrow 1/(2\theta)$ gives that $\gamma(1 + 2\theta) \notin D$, since $M \setminus D$ is closed. \square

8.2.2. Compatibility

Given a barrier F , for which convex functions f is it the case that $tf + F$ is self-concordant for all $t \geq 0$, with parameter independent of t ? This is clearly the case if f is (affine) linear or quadratic in the sense that the third covariant derivative $\nabla^3 f$ vanishes. We now define the more general notion of *compatibility*, which suffices for this, as shown in Proposition 8.2.10 below.

Definition 8.2.8 (Compatibility). Let $D \subseteq M$ be open and convex, let $f, F: D \rightarrow \mathbb{R}$ be convex functions. For $\beta_1, \beta_2 \geq 0$, we say that f is (β_1, β_2) -compatible with F if for all $p \in D$ and $u, v \in T_pM$, one has

$$\begin{aligned} |(\nabla^3 f)_p(u, v, v)| &\leq 2\beta_1 \sqrt{(\nabla^2 F)_p(u, u)} (\nabla^2 f)_p(v, v) \\ &\quad + 2\beta_2 \sqrt{(\nabla^2 F)_p(v, v)} \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(v, v)}. \end{aligned} \tag{8.2.1}$$

8. Interior-point methods on manifolds: the framework

For $\beta \geq 0$, we say that f is β -compatible with F along geodesics if for all $p \in D$ and $v \in T_p M$,

$$|(\nabla^3 f)_p(v, v, v)| \leq 2\beta \sqrt{(\nabla^2 F)_p(v, v)} (\nabla^2 f)_p(v, v). \quad (8.2.2)$$

Clearly, if f is a linear or a convex quadratic function, in the sense that its second or third covariant derivative vanishes, then it is clearly automatically compatible with any convex F . Moreover, any α -self-concordant function is (β_1, β_2) -compatible with itself, for $\beta_1 + \beta_2 = 1/\sqrt{\alpha}$. As we show in Proposition 8.2.10, given a barrier F for a domain D and a convex objective function f , compatibility guarantees that $tf + F$ is self-concordant for all $t \geq 0$, with a parameter independent of t , and hence one can use the path-following method presented in Section 8.2.3 below to optimize f over D . We apply this theory in Chapter 10.

Compatibility along geodesics reduces to the well-known Euclidean notion, see [NN94, Def. 3.2.1] or [Nes18, Def. 5.4.2]. In these works it is also explained how to generalize the notion of compatibility to vector-valued functions f , which is useful for constructing new barriers out of old ones; see [NN94, §5.1.2] or [Nes18, §5.4.6] for details. We do not provide such a generalization here. Clearly, if f is (β_1, β_2) -compatible with F then it is also β -compatible with F along geodesics for $\beta := \beta_1 + \beta_2$. Yet the latter does not imply the former, even in the Euclidean setting.

We may equivalently write Eqs. (8.2.1) and (8.2.2) as follows in terms of the seminorms $\|\cdot\|_{g,p} = \|\cdot\|_{g,p,1}$ induced by the inner products $\langle \cdot, \cdot \rangle_{g,p} = \langle \cdot, \cdot \rangle_{g,p,1}$ defined in Eq. (8.1.2):

$$|(\nabla^3 f)_p(u, v, v)| \leq 2\beta_1 \|u\|_{F,p} \|v\|_{f,p}^2 + 2\beta_2 \|v\|_{F,p} \|u\|_{f,p} \|v\|_{f,p} \quad (8.2.3)$$

and

$$|(\nabla^3 f)_p(v, v, v)| \leq 2\beta \|v\|_{F,p} \|v\|_{f,p}^2. \quad (8.2.4)$$

We now state some basic properties of compatibility. The following result holds analogously for compatibility along geodesics.

Lemma 8.2.9. *Let $D \subseteq M$ be open and convex, $F: D \rightarrow \mathbb{R}$ a convex function, and $\beta \in \mathbb{R}_{\geq 0}^2$.*

- (i) *Let $f: D \rightarrow \mathbb{R}$ be a convex function that is β -compatible with F and let $c \geq 0$. Then cf is β -compatible with F .*
- (ii) *Let $f_1, f_2: D \rightarrow \mathbb{R}$ be two convex functions that are each β -compatible with F . Then their sum $f_1 + f_2$ is β -compatible with F .*

Proof. Property (i) is clear from the definition, as both sides of Eq. (8.2.1) are positively homogeneous in f . To prove property (ii), we note that for every $p \in D$ and $u, v \in T_p M$,

$$\begin{aligned} & |(\nabla^3(f_1 + f_2))_p(u, v, v)| \leq |(\nabla^3 f_1)_p(u, v, v)| + |(\nabla^3 f_2)_p(u, v, v)| \\ & \leq 2\beta_1 \sqrt{(\nabla^2 F)_p(u, u)} (\nabla^2 f_1)_p(v, v) + 2\beta_1 \sqrt{(\nabla^2 F)_p(u, u)} (\nabla^2 f_2)_p(v, v) \\ & \quad + 2\beta_2 \sqrt{(\nabla^2 F)_p(v, v)} \left(\sqrt{(\nabla^2 f_1)_p(u, u)} \sqrt{(\nabla^2 f_1)_p(v, v)} + \sqrt{(\nabla^2 f_2)_p(u, u)} \sqrt{(\nabla^2 f_2)_p(v, v)} \right) \end{aligned}$$

$$\begin{aligned}
 &\leq 2\beta_1\sqrt{(\nabla^2 F)_p(u, u)(\nabla^2 f_1)_p(v, v)} + 2\beta_1\sqrt{(\nabla^2 F)_p(u, u)(\nabla^2 f_2)_p(v, v)} \\
 &+ 2\beta_2\sqrt{(\nabla^2 F)_p(v, v)\sqrt{(\nabla^2 f_1)_p(u, u) + (\nabla^2 f_2)_p(u, u)}\sqrt{(\nabla^2 f_1)_p(v, v) + (\nabla^2 f_2)_p(v, v)}} \\
 &= 2\beta_1\sqrt{(\nabla^2 F)_p(u, u)(\nabla^2(f_1 + f_2))_p(v, v)} \\
 &+ 2\beta_2\sqrt{(\nabla^2 F)_p(v, v)\sqrt{(\nabla^2(f_1 + f_2))_p(u, u)}\sqrt{(\nabla^2(f_1 + f_2))_p(v, v)}}.
 \end{aligned}$$

The first inequality holds by compatibility of f_1 and of f_2 with F , and the second inequality uses the Cauchy-Schwarz inequality. \square

We now show that if a convex function f is compatible with a self-concordant function F (e.g., a barrier), then $tf + F$ is self-concordant for every $t \geq 0$, with a self-concordance constant that is independent of t . We emphasize that it is not necessary for f itself to be self-concordant. The proof is inspired by [NN94, Prop. 3.2.2] in the Euclidean setting. The result holds analogously if we use compatibility and self-concordance along geodesics in the hypothesis and conclusion.

Proposition 8.2.10. *Let $D \subseteq M$ be open and convex and let $f, F: D \rightarrow \mathbb{R}$ be convex functions. Suppose that f is (β_1, β_2) -compatible with F and F is 1-self-concordant. Then $tf + F: D \rightarrow \mathbb{R}$ is α -self-concordant for every $t \geq 0$, with*

$$\alpha := \begin{cases} \frac{4(\beta_2^2 - (\beta_1 - 1)^2)}{\beta_2^2(\beta_2^2 + 4\beta_1)} & \text{if } \beta_2^2 > 2 \max\{\beta_1(\beta_1 - 1), 1 - \beta_1\}, \\ \frac{1}{\max\{\beta_1^2, 1\}} & \text{otherwise.} \end{cases}$$

If in addition F is strongly 1-self-concordant and f has a closed convex extension, then $tf + F: D \rightarrow \mathbb{R}$ is strongly α -self-concordant for every $t \geq 0$.

Proof. We abbreviate $F_t := tf + F$. Clearly, F_t is convex for every $t \geq 0$, so it remains to prove the self-concordance estimate. For any $p \in D$ and $u, v \in T_p M$, using Eqs. (8.1.6) and (8.2.3),

$$\begin{aligned}
 &|(\nabla^3 F_t)_p(u, v, v)| \\
 &\leq t|(\nabla^3 f)_p(u, v, v)| + |(\nabla^3 F)_p(u, v, v)| \\
 &\leq 2t\beta_1\|u\|_{F,p}\|v\|_{f,p}^2 + 2t\beta_2\|v\|_{F,p}\|u\|_{f,p}\|v\|_{f,p} + 2\|u\|_{F,p}\|v\|_{F,p}^2 \\
 &= 2\left(\sqrt{t}\|u\|_{f,p}\left(\sqrt{t}\beta_2\|v\|_{F,p}\|v\|_{f,p}\right) + \|u\|_{F,p}\left(t\beta_1\|v\|_{f,p}^2 + \|v\|_{F,p}^2\right)\right) \\
 &\leq 2\sqrt{t\|u\|_{f,p}^2 + \|u\|_{F,p}^2}\sqrt{\left(\sqrt{t}\beta_2\|v\|_{F,p}\|v\|_{f,p}\right)^2 + \left(t\beta_1\|v\|_{f,p}^2 + \|v\|_{F,p}^2\right)^2} \\
 &= 2\|u\|_{F_t,p}\sqrt{t\beta_2^2\|v\|_{F,p}^2\|v\|_{f,p}^2 + \left(t\beta_1\|v\|_{f,p}^2 + \|v\|_{F,p}^2\right)^2}
 \end{aligned}$$

using the Cauchy-Schwarz inequality in the second-to-last step. To show that F_t is α -self-concordant, by Eq. (8.1.5) it therefore suffices to show that (note we use $\|\cdot\|_{g,p,1}$ rather than $\|\cdot\|_{g,p,\alpha}$!)

$$\sqrt{t\beta_2^2\|v\|_{F,p}^2\|v\|_{f,p}^2 + \left(t\beta_1\|v\|_{f,p}^2 + \|v\|_{F,p}^2\right)^2} \leq \frac{1}{\sqrt{\alpha}}\|v\|_{F_t,p}^2. \quad (8.2.5)$$

8. Interior-point methods on manifolds: the framework

Without loss of generality, we can assume that $\|v\|_{F_t, p}^2 = 1$. Writing $x := \|v\|_{f, p}^2$ and $y := \|v\|_{F, p}^2$, we see that Eq. (8.2.5) holds provided we can prove that

$$\beta_2^2 t x y + (\beta_1 t x + y)^2 \leq \frac{1}{\alpha} \quad (8.2.6)$$

for all $x, y \geq 0$ subject to the constraint $t x + y = 1$. Eliminating t and x using this constraint, the left-hand side can be written as

$$\begin{aligned} q(y) &:= \beta_2^2(1-y)y + (\beta_1(1-y) + y)^2 \\ &= \left((1-\beta_1)^2 - \beta_2^2 \right) y^2 + \left(2\beta_1 - 2\beta_1^2 + \beta_2^2 \right) y + \beta_1^2, \end{aligned}$$

so we wish to show that $q(y) \leq 1/\alpha$ for all $y \in [0, 1]$. Note that $q(y)$ is a quadratic polynomial. We distinguish two cases:

If $(1-\beta_1)^2 < \beta_2^2$, then q is strictly concave and attains its maximum on \mathbb{R} at

$$y_* = \frac{2\beta_1 - 2\beta_1^2 + \beta_2^2}{2(\beta_2^2 - (1-\beta_1)^2)}.$$

Note that $y_* \in (0, 1)$ if and only if

$$0 < 2\beta_1 - 2\beta_1^2 + \beta_2^2 < 2(\beta_2^2 - (1-\beta_1)^2),$$

which is equivalent to

$$\beta_2^2 > 2 \max\{\beta_1(\beta_1 - 1), 1 - \beta_1\}.$$

If $y_* \in (0, 1)$, then the maximum of $q(y)$ on $[0, 1]$ is given by

$$q(y_*) = \frac{(2\beta_1 - 2\beta_1^2 + \beta_2^2)^2}{4(\beta_2^2 - (1-\beta_1)^2)} + \beta_1^2 = \frac{\beta_2^4 + 4\beta_1\beta_2^2}{4(\beta_2^2 - (1-\beta_1)^2)},$$

while otherwise it is attained at the boundary, where $q(0) = \beta_1^2$ and $q(1) = 1$.

If $(1-\beta_1)^2 \geq \beta_2^2$, then $q(y)$ is convex and hence attains its maximum always at the boundary. Summarizing both cases, we find that

$$\max_{y \in [0, 1]} q(y) = \begin{cases} \frac{\beta_2^4 + 4\beta_1\beta_2^2}{4(\beta_2^2 - (1-\beta_1)^2)} & \text{if } (\beta_1 - 1)^2 < \beta_2^2 \text{ and } \beta_2^2 > 2 \max\{\beta_1(\beta_1 - 1), 1 - \beta_1\}, \\ \max\{\beta_1^2, 1\} & \text{otherwise.} \end{cases}$$

The condition of the first case is equivalent to

$$\beta_2^2 > 2 \max\{\beta_1(\beta_1 - 1), 1 - \beta_1\},$$

and hence we have confirmed Eq. (8.2.6). Thus we have proved that $F_t = t f + F$ is an α -self-concordant function on D . Finally, the last claim follows from Lemma 6.5.2 \square

Finally, we construct a self-concordant barrier for the epigraph of any function compatible with a barrier for its domain. This result generalizes the Euclidean result [Nes18, Thm. 5.3.5], which constructs a self-concordant barrier for the open epigraph

$$E_f^\circ := \{(p, t) \in D \times \mathbb{R} : f(p) < t\} \quad (8.2.7)$$

of a self-concordant barrier. As before, it holds analogously if we use the notions along geodesics in the hypothesis and conclusion.

Theorem 8.2.11 (Barriers for epigraphs). *Let $D \subseteq M$ be open and convex and let $f, F: D \rightarrow \mathbb{R}$ be convex functions. Suppose that f is (β_1, β_2) -compatible with F and F is 1-self-concordant. Then, the function*

$$G: E_f^\circ \rightarrow \mathbb{R}, \quad G(p, t) = -\log(t - f(p)) + F(p)$$

defined on the open epigraph E_f° , see Eq. (8.2.7), is convex and α -self-concordant, with

$$\alpha^{-1} := \max\left\{1 + \beta_1^2, \beta_1 + \frac{1}{2}\beta_2^2, \frac{2}{3}\beta_2^2\right\}. \quad (8.2.8)$$

Furthermore, for every $(p, t) \in E_f^\circ$ one has

$$\lambda_{G,\alpha}(p, t)^2 = \frac{\lambda_G(p, t)^2}{\alpha} \leq \frac{1 + \lambda_F(p)^2}{\alpha}. \quad (8.2.9)$$

If in addition F is strongly 1-self-concordant and f has a closed convex extension, then G is strongly α -self-concordant. In particular, if F is a θ -barrier for D and f has a closed convex extension, then G/α is a $(1 + \theta)/\alpha$ -barrier for E_f° .

Proof. We identify $v \in T_{(p,t)}E_f^\circ \cong T_p D \oplus \mathbb{R}$ and write $v = (v_p, v_t)$, with $v_p \in T_p D$ and $v_t \in \mathbb{R}$. Then the differential of G is given by

$$dG_{(p,t)}(v) = -\frac{1}{t - f(p)}(v_t - df_p(v_p)) + dF_p(v_p) \quad (8.2.10)$$

and the Hessian of G by

$$\begin{aligned} & (\nabla^2 G)_{(p,t)}(v, v) & (8.2.11) \\ &= \underbrace{\frac{1}{(t - f(p))^2}(v_t - df_p(v_p))^2}_{=: A_v^2} + \underbrace{\frac{1}{t - f(p)}(\nabla^2 f)_p(v_p, v_p)}_{=: B_v^2} + \underbrace{(\nabla^2 F)_p(v_p, v_p)}_{=: C_v^2}. \end{aligned}$$

The underbraced terms are all non-negative as $t > f(p)$ and both f and F are convex, hence we can write them as squares of real numbers A_v, B_v, C_v . This also shows that G is convex. We now prove that G is self-concordant. The third covariant derivative can be computed as follows: for all $u, v \in T_{(p,t)}E_f^\circ$, we have

$$\begin{aligned} (\nabla^3 G)_{(p,t)}(u, v, v) &= -\frac{2}{(t - f(p))^3}(u_t - df_p(u_p))(v_t - df_p(v_p))^2 \\ &\quad - \frac{2}{(t - f(p))^2}(v_t - df_p(v_p))(\nabla^2 f)_p(u_p, v_p) \\ &\quad - \frac{1}{(t - f(p))^2}(u_t - df_p(u_p))(\nabla^2 f)_p(v_p, v_p) \\ &\quad + \frac{1}{t - f(p)}(\nabla^3 f)_p(u_p, v_p, v_p) + (\nabla^3 F)_p(u_p, v_p, v_p) \\ &= -2A_u A_v^2 - 2A_v \frac{1}{t - f(p)}(\nabla^2 f)_p(u_p, v_p) - A_u B_v^2 \\ &\quad + \frac{1}{t - f(p)}(\nabla^3 f)_p(u_p, v_p, v_p) + (\nabla^3 F)_p(u_p, v_p, v_p). \end{aligned}$$

Now, we have

$$\frac{1}{t - f(p)} (\nabla^2 f)_p(u_p, v_p) \leq B_u B_v$$

by the Cauchy-Schwarz inequality,

$$\begin{aligned} \frac{1}{t - f(p)} |(\nabla^3 f)_p(u_p, v_p, v_p)| &\leq \frac{2}{t - f(p)} \left(\beta_1 \|u\|_{F,p} \|v\|_{f,p}^2 + \beta_2 \|v\|_{F,p} \|u\|_{f,p} \|v\|_{f,p} \right) \\ &= 2 \left(\beta_1 B_v^2 C_u + \beta_2 B_u B_v C_v \right) \end{aligned}$$

by compatibility of f with F as in Eq. (8.2.3), and finally

$$|(\nabla^3 F)_p(u_p, v_p, v_p)| \leq 2C_u C_v^2$$

by 1-self-concordance of F (Eq. (8.1.5)). Combining these estimates, we can upper bound the third covariant derivative of G in absolute value as

$$\begin{aligned} &|(\nabla^3 G)_{(p,t)}(u, v, v)| \\ &\leq 2A_u A_v^2 + 2A_v B_u B_v + A_u B_v^2 + 2 \left(\beta_1 B_v^2 C_u + \beta_2 B_u B_v C_v \right) + 2C_u C_v^2 \\ &= A_u (2A_v^2 + B_v^2) + B_u (2A_v B_v + 2\beta_2 B_v C_v) + C_u (2\beta_1 B_v^2 + 2C_v^2) \\ &\leq \sqrt{A_u^2 + B_u^2 + C_u^2} \sqrt{(2A_v^2 + B_v^2)^2 + (2A_v B_v + 2\beta_2 B_v C_v)^2 + (2\beta_1 B_v^2 + 2C_v^2)^2} \\ &\leq 2\sqrt{(\nabla^2 G)_{(p,t)}(u, u)} \sqrt{\max\{1 + \beta_1^2, \beta_1 + \frac{1}{2}\beta_2^2, \frac{2}{3}\beta_2^2\}} (\nabla^2 G)_{(p,t)}(v, v) \\ &= \frac{2}{\sqrt{\alpha}} \sqrt{(\nabla^2 G)_{(p,t)}(u, u)} (\nabla^2 G)_{(p,t)}(v, v), \end{aligned}$$

where the last inequality holds due to $2xy \leq x^2 + y^2$, as in

$$\begin{aligned} &\frac{1}{4} \left[(2A_v^2 + B_v^2)^2 + (2A_v B_v + 2\beta_2 B_v C_v)^2 + (2\beta_1 B_v^2 + 2C_v^2)^2 \right] \\ &= A_v^4 + \left(\frac{1}{4} + \beta_1^2 \right) B_v^4 + C_v^4 + 2A_v^2 B_v^2 + 2 \left(\beta_1 + \frac{1}{2}\beta_2^2 \right) B_v^2 C_v^2 + 2\beta_2 A_v B_v^2 C_v \\ &= A_v^4 + \left(\frac{1}{4} + \beta_1^2 \right) B_v^4 + C_v^4 + 2A_v^2 B_v^2 + 2 \left(\beta_1 + \frac{1}{2}\beta_2^2 \right) B_v^2 C_v^2 + 2 \left(\frac{\sqrt{3}}{2} B_v^2 \right) \left(\frac{2}{\sqrt{3}} \beta_2 A_v C_v \right) \\ &\leq A_v^4 + \left(\frac{1}{4} + \beta_1^2 \right) B_v^4 + C_v^4 + 2A_v^2 B_v^2 + 2 \left(\beta_1 + \frac{1}{2}\beta_2^2 \right) B_v^2 C_v^2 + \frac{3}{4} B_v^4 + \frac{4}{3} \beta_2^2 A_v^2 C_v^2 \\ &= A_v^4 + \left(1 + \beta_1^2 \right) B_v^4 + C_v^4 + 2A_v^2 B_v^2 + 2 \left(\beta_1 + \frac{1}{2}\beta_2^2 \right) B_v^2 C_v^2 + 2\frac{2}{3}\beta_2^2 A_v^2 C_v^2 \\ &\leq \max\{1 + \beta_1^2, \beta_1 + \frac{1}{2}\beta_2^2, \frac{2}{3}\beta_2^2\} \left(A_v^2 + B_v^2 + C_v^2 \right)^2. \end{aligned}$$

We conclude that G is indeed α -self-concordant with α as in Eq. (8.2.8).

Next, we prove the bound on the differential. Using Eq. (8.2.10) and with A_v, B_v as in Eq. (8.2.11), we have

$$\begin{aligned} |dG_{(p,t)}(v)| &\leq A_v + |dF_p(v_p)| \leq A_v + \lambda_F(p) C_v \\ &\leq \sqrt{1 + \lambda_F(p)^2} \sqrt{A_v^2 + C_v^2} \leq \sqrt{1 + \lambda_F(p)^2} \sqrt{(\nabla^2 G)_{(p,t)}(v, v)}, \end{aligned}$$

by definition of the Newton decrement and the Cauchy-Schwarz inequality. Thus we find that

$$\lambda_{G,\alpha}(p, t) \leq \sqrt{\frac{1 + \lambda_F(p)^2}{\alpha}}.$$

which establishes Eq. (8.2.9).

Finally, if F is strongly 1-self-concordant, hence closed convex on D , and if f has a closed convex extension then it is easy to see that G is closed convex on E_f° , using that $(s, t) \mapsto -\log(t - s)$ is closed convex on $\{(s, t) \in \mathbb{R}^2 : s < t\}$. \square

In particular, we can apply this construction to any self-concordant function:

Corollary 8.2.12. *Let $D \subseteq M$ be open and convex and let $f: D \rightarrow \mathbb{R}$ be 1-self-concordant. Then $g(p, t) = -\log(t - f(p)) + f(p)$ is a convex and 1-self-concordant function on the open epigraph E_f° of f , see Eq. (8.2.7). It satisfies $\lambda_g(p, t) \leq \sqrt{1 + \lambda_f(p)^2}$ for all $(p, t) \in E_f^\circ$. If f is strongly self-concordant, so is g . In particular, if f is a θ -barrier, g is a $(1 + \theta)$ -barrier for E_f° .*

To end this section, we provide a variant of the above barrier for level sets of a convex function which does not use the notion of compatibility, but has a parameter that depends on the variation of the function. For a convex function $f: M \rightarrow \mathbb{R}$ and $\eta \in \mathbb{R}$ for which there is $p \in M$ with $f(p) < \eta$, the open level set $\mathcal{L}_{f,\eta}^\circ \subseteq M$ is defined by

$$\mathcal{L}_{f,\eta}^\circ = \{p \in M \mid f(p) < \eta\}. \quad (8.2.12)$$

Define the logarithmic barrier $F_\eta: \mathcal{L}_{f,\eta}^\circ \rightarrow \mathbb{R}$ by

$$F_\eta(p) = -\log(\eta - f(p)) \quad (p \in \mathcal{L}_{f,\eta}^\circ). \quad (8.2.13)$$

The logarithmic barrier is convex and has bounded Newton decrements as follows.

Lemma 8.2.13. *The function $F = F_\eta$ defined in Eq. (8.2.13) is smooth, closed convex, and satisfies*

$$dF_p(u)^2 \leq (\nabla^2 F)_p(u, u) \quad (u \in T_p M, p \in \mathcal{L}_{f,\eta}^\circ). \quad (8.2.14)$$

Proof. Let $\omega(p) := \eta - f(p) > 0$. Then we have

$$dF_p(u) = \frac{df_p(u)}{\omega(p)}, \quad (\nabla^2 F)_p(u, u) = \frac{(\nabla^2 f)_p(u, u)}{\omega(p)} + \frac{df_p(u)^2}{\omega(p)^2}. \quad (8.2.15)$$

Then by convexity of f , $(\nabla^2 f)_p(u, u) \geq 0$ and hence F is convex, and satisfies $(\nabla^2 F)_p(u, u) \geq dF_p(u)^2$.

The closedness of F is seen as follows: Consider a sequence (p_k, z_k) in the epigraph of F , that converges to $(p_\infty, z_\infty) \in M \times \mathbb{R}$. Note that f is smooth on M , and hence so is F on $\mathcal{L}_{f,\eta}^\circ$. By continuity of f , $\mathcal{L}_{f,\eta}^\circ$ is open, hence disjoint from its boundary in M . Therefore any boundary point q of $\mathcal{L}_{f,\eta}^\circ$ satisfies $f(q) \geq \eta$. Therefore, it is impossible for p_∞ to belong to the boundary of $\mathcal{L}_{f,\eta}^\circ$: that would imply $f(p_\infty) \geq \eta$, which would imply $z_\infty \geq \infty$. Hence $p_\infty \in \mathcal{L}_{f,\eta}^\circ$, and $F(z_\infty) = \lim_{k \rightarrow \infty} F(p_k) \leq \lim_{k \rightarrow \infty} z_k = z_\infty$. \square

8. Interior-point methods on manifolds: the framework

If an α -self-concordant function F satisfies Eq. (8.2.14), then F/α is an α -barrier. The following is an extension of [Nes18, Thm. 5.1.4] to our setting; the result below is originally due to H. Hirai [Hir22b]:

Theorem 8.2.14 (Barriers for level sets). *Suppose that $f: M \rightarrow \mathbb{R}$ is α -self-concordant. Then $F_\eta: \mathcal{L}_{f,\eta}^\circ \rightarrow \mathbb{R}$ is α' -self-concordant for*

$$\alpha' = \frac{4(\eta - f^*)/\alpha + 1}{(2(\eta - f^*)/\alpha + 1)^2} \quad (8.2.16)$$

where $f^* := \inf_{x \in M} f(x)$. In particular, F_η/α' is an $O((\eta - f^*)/\alpha)$ -barrier for $\mathcal{L}_{f,\eta}^\circ$.

When only considering self-concordance along geodesics, the constant α' can be taken as $\alpha/((\eta - f^*) + \alpha)$, which is exactly what is proven in [Nes18, Thm. 5.1.4]. For self-concordance, however, a little modification is required, which leads to a weaker constant.

Proof. Our starting point is Eq. (8.2.15), where we recall that $\omega(p) = \eta - f(p)$. Since $df_p(u)^2 = (df_p \otimes df_p)(u, u)$, and $(\nabla_v(df \otimes df))_p(u, u) = ((\nabla_v df)_p \otimes df_p + df_p \otimes (\nabla_v df)_p)(u, u) = 2df_p(u)(\nabla^2 f)_p(u, v)$, the covariant derivative of $\nabla^2 F$ is given by (suppressing p 's for convenience)

$$\nabla^3 F(v, u, u) = \frac{\nabla^3 f(v, u, u)}{\omega} + \frac{df(v)\nabla^2 f(u, u)}{\omega^2} + \frac{2df(u)\nabla^2 f(u, v)}{\omega^2} + \frac{2df(v)df(u)^2}{\omega^3}. \quad (8.2.17)$$

Hence we have

$$\begin{aligned} |\nabla^3 F(v, u, u)| &\leq \frac{2\sqrt{\nabla^2 f(v, v)}\nabla^2 f(u, u)}{\sqrt{\alpha}\omega} + \frac{|df(v)|\nabla^2 f(u, u)}{\omega^2} \\ &\quad + \frac{2|df(u)|\sqrt{\nabla^2 f(v, v)}\sqrt{\nabla^2 f(u, u)}}{\omega^2} + \frac{2|df(v)|df(u)^2}{\omega^3}. \end{aligned}$$

Define τ_1, τ, ξ_1, ξ by

$$\tau_1 := \sqrt{\nabla^2 f(v, v)}/\omega, \quad \tau := \sqrt{\nabla^2 f(u, u)}/\omega, \quad \xi_1 := |df(v)|/\omega, \quad \xi := |df(u)|/\omega.$$

Then we have

$$\frac{|\nabla^3 F(v, u, u)|}{2\sqrt{\nabla^2 F(v, v)}\nabla^2 F(u, u)} \leq \frac{(1/\sqrt{\alpha})\omega^{1/2}\tau_1\tau^2 + (1/2)\xi_1\tau^2 + \xi\tau_1\tau + \xi_1\xi^2}{(\tau_1^2 + \xi_1^2)^{1/2}(\tau^2 + \xi^2)}. \quad (8.2.18)$$

We bound the right-hand side as follows. By homogeneity, we may consider the optimization problem:

$$\text{maximize } (1/\sqrt{\alpha})\omega^{1/2}\tau_1\tau^2 + (1/2)\xi_1\tau^2 + \xi\tau_1\tau + \xi_1\xi^2 \quad \text{s.t. } \tau_1^2 + \xi_1^2 = 1, \quad \tau^2 + \xi^2 = 1.$$

For fixed (τ, ξ) , optimizing with respect to (τ_1, ξ_1) is a linear optimization over the unit circle, and the optimum is attained at

$$\frac{((1/\sqrt{\alpha})\omega^{1/2}\tau^2 + \xi\tau, (1/2)\tau^2 + \xi^2)}{\sqrt{((1/\sqrt{\alpha})\omega^{1/2}\tau^2 + \xi\tau)^2 + ((1/2)\tau^2 + \xi^2)^2}}.$$

Then the problem reduces to

$$\text{maximize } \sqrt{((1/\sqrt{\alpha})\omega^{1/2}\tau^2 + \xi\tau)^2 + ((1/2)\tau^2 + \xi^2)^2} \text{ s.t. } \tau^2 + \xi^2 = 1.$$

This optimization problem can be solved using the method of Lagrange multipliers. For convenience set $c = \sqrt{\omega/\alpha}$, and define $q(\tau, \xi) = (\sqrt{\omega/\alpha}\tau^2 + \xi\tau)^2 + (\tau^2/2 + \xi^2)^2$. The system of equations

$$\partial_\tau q(\tau, \xi) = \mu\tau, \quad \partial_\xi q(\tau, \xi) = \mu\xi, \quad \tau^2 + \xi^2 = 1, \quad \mu \in \mathbb{R}$$

has six solutions (τ, ξ, μ) , given by

$$(0, \pm 1, 4), \quad \frac{1}{\sqrt{4c^2 + 1}}(2c, 1, 16c^4 + 16c^2 + 4), \quad \frac{1}{\sqrt{4c^2 + 1}}(-2c, -1, 16c^4 + 16c^2 + 4), \\ \frac{1}{\sqrt{4c^2 + 9}}(3, -2c, 16c^2 + 9), \quad \frac{1}{\sqrt{4c^2 + 9}}(-3, 2c, 16c^2 + 9)$$

and the largest value attained of $q(\tau, \xi)$ attained at any of these points is $(2c^2 + 1)^2/(4c^2 + 1)$. Therefore, the right-hand side of Eq. (8.2.18) is at most

$$\sqrt{\frac{(2(\omega/\alpha) + 1)^2}{4(\omega/\alpha) + 1}}.$$

In other words, this gives that $\alpha' = (4(\omega/\alpha) + 1)/(2(\omega/\alpha) + 1)^2$ is a suitable self-concordance constant at p . Taking the maximum over $p \in \mathcal{L}_{f,\eta}^\circ$ yields the choice of α' in Eq. (8.2.16). \square

8.2.3. Path-following method

We now discuss a path-following method for objectives which are compatible with a barrier. To this end, we consider the approach of [NN94, Ch. 3]. Their Euclidean framework is rather general, and deals with *self-concordant families*. We specialize to self-concordant families generated by a barrier, and generalize the corresponding path-following method to the Riemannian setting. The goal is to minimize a convex objective function f over an open convex domain D , that is, to find $p \in D$ such that $f(p) \approx \inf_{q \in D} f(q)$. The running assumption we shall make is that we have a barrier F for the domain D such that the function

$$F_t := tf + F: D \rightarrow \mathbb{R}$$

is α -self-concordant for all $t \geq 0$, with a parameter α that is independent of t . One way to guarantee this is to assume that f is compatible with F , as shown before in Proposition 8.2.10.

The basic idea of the path-following method is as follows (as explained previously in Chapter 4). The algorithm keeps track of two pieces of data, a point p in the domain D and a time parameter t . The initial data to the algorithm is specified by a point $p_{-1} \in D$ such that $\lambda_{F,\alpha}(p_{-1})$ is small. We then choose a time parameter $t_0 > 0$ such that we are in the quadratic convergence regime for Newton's method for F_{t_0} as determined by Theorem 8.1.17, say $\lambda_{F_{t_0},\alpha}(p_{-1}) < \lambda_* = 1 - 1/\sqrt{2}$. Such initial data can be obtained for instance by using the damped Newton method of Theorem 8.1.18, or in the Euclidean setting by a similar (reverse) path-following method. We then iterate the following procedure for $k = 0, 1, 2, \dots$:

8. Interior-point methods on manifolds: the framework

- (i) Update p_{k-1} to $p_k \in D$ by taking one Newton step with respect to F_{t_k} , so that $\lambda_{F_{t_k}, \alpha}(p_{k+1})$ becomes smaller.
- (ii) Increase t_k to some t_{k+1} by a constant factor, such that $\lambda_{F_{t_{k+1}}, \alpha}(p_k) < \lambda_*$ still holds.

Throughout the algorithm, p_k will be an approximate minimizer of F_{t_k} . One can also show that if t_k is large enough, approximate minimizers of F_{t_k} are approximate minimizers of f .

We first determine by what factor one can increase t while keeping the Newton decrement below some threshold. The following result is a translation of [NN94, Thm. 3.1.1] to our setting. Note that here, we do not assume that $tf + F$ is self-concordant.

Lemma 8.2.15. *Let $D \subseteq M$ be open and convex, let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics, and let $f: D \rightarrow \mathbb{R}$ be a convex function. Furthermore, let $t, t', \alpha, c > 0$ and $p \in D$ be such that*

$$\left(1 + \frac{\sqrt{\theta}}{c\sqrt{\alpha}}\right) \left| \log \frac{t'}{t} \right| \leq 1 - \frac{\lambda_{F_t, \alpha}(p)}{c}.$$

Then $\lambda_{F_t, \alpha}(p) \leq c$ implies that $\lambda_{F_{t'}, \alpha}(p) \leq c$.

Proof. Let $p \in D$. Throughout the proof, all derivatives of functions defined on M will be taken at the point p , hence we shall omit the subscript. We will assume that $t' \geq t$, but the proof for $t' \leq t$ is analogous. For every $0 \neq u \in T_p M$, define a function $\phi_u: [t, t'] \rightarrow \mathbb{R}$ by

$$\phi_u(s) = \frac{dF_s(u)}{\sqrt{\nabla^2 F_s(u, u)}}.$$

To prove the lemma, it suffices to show that $|\phi_u(t')| \leq c\sqrt{\alpha}$ for all $u \neq 0$. Since $\phi_{-u} = -\phi_u$, we may assume without loss of generality that $\phi_u(t') \geq 0$. We first compute the derivative of ϕ_u :

$$\begin{aligned} \partial_s \phi_u(s) &= \frac{df(u)}{\sqrt{\nabla^2 F_s(u, u)}} - \frac{1}{2} \frac{dF_s(u) \cdot \nabla^2 f(u, u)}{(\nabla^2 F_s(u, u))^{3/2}} \\ &= \frac{1}{s} \phi_u(s) - \frac{1}{s} \frac{dF(u)}{\sqrt{\nabla^2 F_s(u, u)}} - \frac{1}{2} \frac{dF_s(u) \cdot \nabla^2 f(u, u)}{(\nabla^2 F_s(u, u))^{3/2}} \\ &= \frac{1}{2s} \phi_u(s) - \frac{1}{s} \frac{dF(u)}{\sqrt{\nabla^2 F_s(u, u)}} + \frac{1}{2s} \frac{dF_s(u) \cdot \nabla^2 F(u, u)}{(\nabla^2 F_s(u, u))^{3/2}} \\ &= \frac{1}{2s} \phi_u(s) \left(1 + \frac{\nabla^2 F(u, u)}{\nabla^2 F_s(u, u)}\right) - \frac{1}{s} \frac{dF(u)}{\sqrt{\nabla^2 F_s(u, u)}}. \end{aligned}$$

Let t_0 be the largest $s \in [t, t']$ such that $\phi_u(t_0) = 0$; if such an s does not exist, then set $t_0 = t$. Let $t^* \in [t_0, t']$ be such that $\phi_u(t^*)$ is maximal over this interval, and set $\phi_u^* = \phi_u(t^*)$. Then,

$$\phi_u^* = \phi_u(t_0) + \int_{t_0}^{t^*} \partial_s \phi(s) ds$$

$$\begin{aligned}
 &\leq \phi_u(t_0) + \int_{t_0}^{t^*} \left[\frac{1}{2s} \phi_u(s) \left(1 + \frac{\nabla^2 F(u, u)}{\nabla^2 F_s(u, u)} \right) + \frac{1}{s} \frac{|dF(u)|}{\sqrt{\nabla^2 F_s(u, u)}} \right] ds \\
 &\leq |\phi_u(t_0)| + \int_{t_0}^{t^*} \left[\frac{1}{s} \phi_u(s) + \frac{1}{s} \sqrt{\theta} \right] ds \\
 &\leq |\phi_u(t)| + (\phi_u^* + \sqrt{\theta}) \log \frac{t^*}{t_0};
 \end{aligned}$$

the second inequality follows since $\nabla^2 F_s \geq \nabla^2 F$ as f is convex and using that F is a θ -barrier; the last inequality is ensured by our choice of t_0 . Using $|\phi_u(t)| \leq \sqrt{\alpha} \lambda_{F_t, \alpha}(p)$, we obtain

$$\phi_u^* \left(1 - \log \frac{t^*}{t_0} \right) \leq \sqrt{\alpha} \lambda_{F_t, \alpha}(p) + \sqrt{\theta} \log \frac{t^*}{t_0}, \quad (8.2.19)$$

On the other hand, since $t \leq t_0 \leq t^* \leq t'$, our assumption implies that

$$\left(1 + \frac{\sqrt{\theta}}{c\sqrt{\alpha}} \right) \log \frac{t^*}{t_0} \leq \left(1 + \frac{\sqrt{\theta}}{c\sqrt{\alpha}} \right) \left| \log \frac{t'}{t} \right| \leq 1 - \frac{\lambda_{F_t, \alpha}(p)}{c},$$

or equivalently

$$\sqrt{\alpha} \lambda_{F_t, \alpha}(p) + \sqrt{\theta} \log \frac{t^*}{t_0} \leq c\sqrt{\alpha} \left(1 - \log \frac{t^*}{t_0} \right). \quad (8.2.20)$$

Combining Eqs. (8.2.19) and (8.2.20) gives $\phi_u^* \leq c\sqrt{\alpha}$, implying that $|\phi_u(t')| \leq c\sqrt{\alpha}$ as desired. \square

We now show that for large $t > 0$, approximate minimizers of F_t correspond to approximate minimizers of f . The proposition and proof we give below are adapted from [NN94, Prop. 3.2.4].

Proposition 8.2.16. *Let $D \subseteq M$ be open and convex, let $F: D \rightarrow \mathbb{R}$ be a θ -barrier along geodesics for D , and let $f: D \rightarrow \mathbb{R}$ be a smooth convex function which has a closed convex extension. For some fixed $t > 0$, suppose that $F_t := tf + F$ is α -self-concordant along geodesics for some $\alpha > 0$ and that it is bounded from below. Then for every $p \in D$ such that $\lambda_{F_t, \alpha}(p) < \frac{1}{3}$, we have*

$$f(p) - \inf_{q \in D} f(q) \leq \frac{2\theta + \alpha\rho(\lambda_{F_t, \alpha}(p))}{t},$$

where we recall from Eq. (8.1.17) that $\rho(r) = -r - \log(1 - r)$.

Proof. By Lemma 6.5.2, F_t is closed convex and hence strongly α -self-concordant along geodesics. Because its Hessian is positive definite and we have $\lambda_{F_t, \alpha}(p) < 1$, Proposition 8.1.14 implies that F_t attains its minimum at a unique minimizer $p_{t,*} \in D$ and moreover

$$F_t(p) - F_t(p_{t,*}) \leq \alpha\rho(\lambda_{F_t, \alpha}(p)). \quad (8.2.21)$$

Furthermore, Lemma 8.1.15 shows that if $u \in T_p M$ is such that $\text{Exp}_p(u) = p_{t,*}$, then

$$\|u\|_{F_t, p, \alpha} \leq \frac{\lambda_{F_t, \alpha}(p)}{1 - \lambda_{F_t, \alpha}(p)} < \frac{1}{2}$$

8. Interior-point methods on manifolds: the framework

where the last inequality follows from $\lambda_{F_t, \alpha}(p) < \frac{1}{3}$. Using Corollary 8.1.11, we obtain that

$$\text{Exp}_{p_{t,*}}(v) = \text{Exp}_p(2u) \in D,$$

where $v = \tau_{\gamma, 1}u$ is the parallel transport of u from p to $p_{t,*}$ along the geodesic $\gamma(t) := \text{Exp}_p(tu)$. By Proposition 8.2.4, it follows that

$$dF_{p_{t,*}}(v) \leq \theta$$

and hence, using convexity of F and $\text{Exp}_{p_{t,*}}(v) = p$,

$$F(p_{t,*}) - F(p) \leq -dF_{p_{t,*}}(-v) = dF_{p_{t,*}}(v) \leq \theta. \quad (8.2.22)$$

Together, Eqs. (8.2.21) and (8.2.22) then show that

$$\begin{aligned} f(p) &= \frac{F_t(p) - F(p)}{t} \\ &\leq \frac{F_t(p_{t,*}) + \alpha\rho(\lambda_{F_t, \alpha}(p)) - F(p)}{t} \\ &= f(p_{t,*}) + \frac{F(p_{t,*}) - F(p) + \alpha\rho(\lambda_{F_t, \alpha}(p))}{t} \\ &\leq f(p_{t,*}) + \frac{\theta + \alpha\rho(\lambda_{F_t, \alpha}(p))}{t}. \end{aligned} \quad (8.2.23)$$

We will now give an upper bound on $f(p_{t,*}) - f(q)$ for every $q \in D$. Let $v \in T_{p_{t,*}}M$ be such that $\text{Exp}_{p_{t,*}}(v) = q$. Using the convexity of f , the fact that $p_{t,*}$ is a minimizer of F_t , and Proposition 8.2.4 (in this order) gives

$$f(p_{t,*}) - f(q) \leq -df_{p_{t,*}}(v) = \frac{dF_{p_{t,*}}(v)}{t} \leq \frac{\theta}{t}.$$

Combining this with Eq. (8.2.23) and optimizing over $q \in D$ gives the desired bound. \square

We now come to the main result of this section, giving a path-following method which converges to a minimizer of the objective, generalizing [NN94, Prop. 3.2.4] to our setting.

Theorem 8.2.17. *Let $D \subseteq M$ be an open, convex, and bounded domain. Let $F: D \rightarrow \mathbb{R}$ be a θ -barrier for D , and let $f: D \rightarrow \mathbb{R}$ be a smooth convex function with a closed convex extension. Let $\alpha > 0$ be such that $F_t := tf + F$ is α -self-concordant for all $t \geq 0$. Choose $1 > \lambda^{(1)} > \lambda^{(2)} > 0$ such that $\left(\frac{\lambda^{(1)}}{1-\lambda^{(1)}}\right)^2 \leq \lambda^{(2)} < \frac{1}{3}$; a suitable choice is given by $\lambda^{(1)} = \frac{1}{4}$, $\lambda^{(2)} = \frac{1}{9}$. Finally, let $p \in D$ be given such that $\lambda_{F, \alpha}(p) < \lambda^{(1)}$, and assume that p is not a minimizer of f . Define a sequence of time parameters*

$$t_0 = \frac{\sqrt{\alpha}\lambda^{(1)} - \lambda_F(p)}{\|df_p\|_{F,p}^*}, \quad t_\ell = t_0 \cdot \exp\left(\ell \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{\theta/\alpha}}\right) \text{ for } \ell = 0, 1, 2, \dots,$$

and a sequence of points

$$p_{-1} = p, \quad p_\ell = (p_{\ell-1})_{F_{t_\ell}, +} \text{ for } \ell = 0, 1, 2, \dots.$$

i.e., p_ℓ is the Newton iterate of $p_{\ell-1}$ with respect to F_{t_ℓ} . Then this sequence is well-defined, in the sense that $p_\ell \in D$ for all $\ell \geq 0$, and it satisfies

$$f(p_\ell) - \inf_{q \in D} f(q) \leq \frac{2(\theta + \alpha)}{t_\ell} = \frac{2(\theta + \alpha) \|df_p\|_{F,p}^*}{\sqrt{\alpha}\lambda^{(1)} - \lambda_F(p)} \exp\left(-\ell \cdot \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{\theta/\alpha}}\right).$$

Proof. By the assumptions on f and strong self-concordance of F , we see from Lemma 6.5.2 that F_t is strongly α -self-concordant on D for all $t \geq 0$. We shall prove by induction on ℓ that for every $\ell \geq 0$, we have $p_\ell \in D$ and

$$\lambda_{F_{t_\ell}, \alpha}(p_{\ell-1}) \leq \lambda^{(1)}, \quad \lambda_{F_{t_\ell}, \alpha}(p_\ell) \leq \lambda^{(2)}.$$

Let us first check that $\lambda_{F_{t_0}, \alpha}(p_{-1}) = \lambda_{F_{t_0}, \alpha}(p) \leq \lambda^{(1)}$. For every $u \neq 0$, we have

$$\begin{aligned} |d(F_{t_0})_p(u)| &\leq t_0 |df_p(u)| + |dF_p(u)| \\ &= (\sqrt{\alpha}\lambda^{(1)} - \lambda_F(p)) \frac{|df_p(u)|}{\|df_p\|_{F,p}^*} + |dF_p(u)| \\ &\leq (\sqrt{\alpha}\lambda^{(1)} - \lambda_F(p)) \|u\|_{F,p} + \|dF_p\|_{F,p}^* \|u\|_{F,p} \\ &= \sqrt{\alpha}\lambda^{(1)} \|u\|_{F,p} \\ &\leq \sqrt{\alpha}\lambda^{(1)} \|u\|_{F_{t_0}, p}, \end{aligned}$$

hence $\|d(F_{t_0})_p\|_{F_{t_0}, p}^* \leq \sqrt{\alpha}\lambda^{(1)}$, which is equivalent to $\lambda_{F_{t_0}, \alpha}(p) \leq \lambda^{(1)}$. Next, if $\lambda_{F_{t_\ell}, \alpha}(p_{\ell-1}) \leq \lambda^{(1)}$ for some $\ell \geq 0$, then by applying Theorem 8.1.16, we find that the Newton iterate p_ℓ is in D and satisfies

$$\lambda_{F_{t_\ell}, \alpha}(p_\ell) \leq \left(\frac{\lambda^{(1)}}{1 - \lambda^{(1)}}\right)^2 \leq \lambda^{(2)}.$$

Lastly, it remains to verify that if $\lambda_{F_{t_\ell}, \alpha}(p_\ell) \leq \lambda^{(2)}$ for some $\ell \geq 0$, then $\lambda_{F_{t_{\ell+1}}, \alpha}(p_\ell) \leq \lambda^{(1)}$. The t_ℓ are chosen exactly so that

$$\left(1 + \frac{\sqrt{\theta}}{\lambda^{(1)}\sqrt{\alpha}}\right) \left|\log \frac{t_\ell}{t_{\ell+1}}\right| = \left(1 + \frac{\sqrt{\theta}}{\lambda^{(1)}\sqrt{\alpha}}\right) \left(\frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{\theta/\alpha}}\right) = 1 - \frac{\lambda^{(2)}}{\lambda^{(1)}}.$$

We conclude that $\lambda_{F_{t_{\ell+1}}, \alpha}(p_\ell) \leq \lambda^{(1)}$ by Lemma 8.2.15. Lastly, the bound on $f(p_1) - \inf_{q \in D} f(q)$ follows from Proposition 8.2.16, where we use that $\lambda^{(2)} < \frac{1}{3}$ and $\rho(\frac{1}{3}) \approx 0.072 \leq 2$. \square

We end with a simple but useful lemma to upper bound the quantity $\|df_p\|_{F,p}^*$.

Lemma 8.2.18. *Let $p \in D$, and $f, F: D \rightarrow \mathbb{R}$ be such that f is convex and F is strongly 1-self-concordant on D . Then*

$$\|df_p\|_{F,p}^* \leq \sup_{q \in D} f(q) - f(p) \leq \sup_{q \in D} f(q) - \inf_{q \in D} f(q).$$

8. Interior-point methods on manifolds: the framework

Proof. By Corollary 8.1.8, the Dikin ellipsoid $B := B_{\mathbb{F},p}^{\circ}(1)$ of radius 1 is contained in D . Then the convexity of f gives

$$\begin{aligned} \|df_p\|_{\mathbb{F},p}^* &= \sup_{\substack{u \in T_p M \\ \|u\|_{\mathbb{F},p} < 1}} |df_p(u)| = \sup_{\substack{u \in T_p M \\ \|u\|_{\mathbb{F},p} < 1}} df_p(u) \\ &\leq \sup_{\substack{u \in T_p M \\ \|u\|_{\mathbb{F},p} < 1}} f(\text{Exp}_p(u)) - f(p) = \sup_{q \in B} f(q) - f(p), \end{aligned}$$

which is at most $\sup_{q \in D} f(q) - f(p)$ as $B \subseteq D$. □

9. Self-concordance of the squared distance in non-positive curvature

In this chapter we discuss self-concordance of the squared distance function to a point. In Section 9.1 we recall some useful formulas that apply to arbitrary Hadamard manifolds. In Section 9.2 we focus on the space $\text{PD}(n)$ of positive-definite complex $n \times n$ matrices and prove that the distance squared to any point is self-concordant. This relies on explicit computations of higher covariant derivatives. Next, in Section 9.3 we use these same formulas to deduce stronger self-concordance estimates in the case of hyperbolic space \mathbb{H}^n , and use these to construct a barrier for the distance function rather than its square; all this generalizes readily to the model spaces of arbitrary constant negative curvature.

9.1. Hadamard manifolds

Let M be a Hadamard manifold, i.e., a simply-connected geodesically-complete Riemannian manifold with non-positive sectional curvature (cf. Section 6.3). Fix $p_0 \in M$ and consider the function that computes the *squared distance* to the point p_0 , that is,

$$f: M \rightarrow \mathbb{R}, \quad f(p) = d(p, p_0)^2.$$

Then it is known that f is 2-strongly convex (which follows from variational principles for the energy of a curve, cf. [Lee18, Thm. 10.22]). In fact, this is a defining property of the more general class of $\text{CAT}(0)$ -spaces, see Theorem 6.2.2 and [BH13]. It will also be useful to consider the distance to p_0 ,

$$g: M \rightarrow \mathbb{R}, \quad g(p) = d(p, p_0),$$

which is still convex. The following lemma summarizes well-known properties of these functions.

Lemma 9.1.1. *Let M be a Hadamard manifold, let $p_0 \in M$, and define $f, g: M \rightarrow \mathbb{R}$ by $f(p) = d(p, p_0)^2$ and $g(p) = d(p, p_0)$. Then f is 2-strongly convex and g is convex. For every $p \neq p_0$, g is smooth at p , and the differentials and Hessians satisfy*

$$df_p = 2g(p)dg_p = -2 \langle \text{Exp}_p^{-1}(p_0), \cdot \rangle_p, \quad (9.1.1)$$

$$\nabla^2 f = 2g\nabla^2 g + 2 dg \otimes dg \geq 2 dg \otimes dg = \frac{df \otimes df}{2f}. \quad (9.1.2)$$

Proof. The strong convexity of f and convexity of g hold on any $\text{CAT}(0)$ -space [BH13, Cor. II.2.5]. Whenever $p \neq p_0$, $f(p) \neq 0$ and hence $g = \sqrt{f}$ is smooth at p . By the

This chapter is adapted from [HNW23].

9. Self-concordance of the squared distance in non-positive curvature

chain rule, $df = 2g dg$. To compute these, note that g is 1-Lipschitz by the triangle inequality, so $|dg_p(u)| \leq \|u\|_p$ for all $u \in T_p M$. But since the geodesic from p in the direction $\text{Exp}_p^{-1}(p_0)$ has constant speed and reaches p_0 at time 1, it follows that

$$dg_p(\text{Exp}_p^{-1}(p_0)) = -g(p).$$

As $\|\text{Exp}_p^{-1}(p_0)\|_p = g(p)$, an application of the Cauchy–Schwarz inequality

$$\begin{aligned} g(p) &= |dg_p(\text{Exp}_p^{-1}(p_0))| = |\langle (\text{grad } g)_p, \text{Exp}_p^{-1}(p_0) \rangle| \\ &\leq \|(\text{grad } g)_p\|_p \|\text{Exp}_p^{-1}(p_0)\|_p \leq \|\text{Exp}_p^{-1}(p_0)\|_p \end{aligned}$$

holds with equality. It follows that $(\text{grad } g)_p = -g(p)^{-1} \text{Exp}_p^{-1}(p_0)$ and $dg_p = -g(p)^{-1} \langle \text{Exp}_p^{-1}(p_0), \cdot \rangle$, and $df_p = -2 \langle \text{Exp}_p^{-1}(p_0), \cdot \rangle_p$. We finally derive the formulas for the Hessians. Applying the product rule to $df = 2g dg$ yields

$$(\nabla^2 f)_p = 2g(p)(\nabla^2 g)_p + 2 dg_p \otimes dg_p,$$

The lower bound in Eq. (9.1.2) follows since $(\nabla^2 g)_p \geq 0$, as a consequence of the convexity of g . \square

Corollary 9.1.2. *The Newton decrement of $f(p) = d(p, p_0)^2$ is given by $\lambda_f(p) = \sqrt{2} d(p, p_0)$.*

Proof. Recall the variational characterization of the Newton decrement (Eq. (8.1.20)):

$$\lambda_f(p) = \min\{\lambda \geq 0 : df_p \otimes df_p \leq \lambda^2 (\nabla^2 f)_p\}.$$

Thus, $\lambda_f \leq \sqrt{2}f$ by Eq. (9.1.2). As g is linear in the direction $\text{Exp}_p^{-1}(p_0)$, its Hessian vanishes in this direction and so we in fact have equality, by the first equality in Eq. (9.1.2). \square

We use Lemma 9.1.1 to prove the following result, which is used later to prove Theorem 9.3.7.

Lemma 9.1.3. *Let $\Psi: M \times \mathbb{R} \times \mathbb{R}_{>0} \rightarrow \mathbb{R}$ be the function defined by*

$$\Psi(p, R, S) = R - S^{-1} d(p, p_0)^2.$$

Then Ψ is concave, with Hessian given by

$$\nabla^2 \Psi = -\frac{2(S^{-1}g dS - dg)^{\otimes 2} + (\nabla^2 f - 2dg \otimes dg)}{S} \leq 0,$$

where f, g are as in Lemma 9.1.1, dS is the differential of the projection $(p, R, S) \mapsto S$, and we write dg for the differential of $(p, R, S) \mapsto g(p)$ by a slight abuse of notation. Moreover, for $u = (u_p, u_R, u_S)$ and $w = (w_p, w_R, w_S)$ tangent vectors at (p, R, S) , one has

$$\nabla^3 \Psi(w, u, u) = -2 \frac{u_S}{S} \nabla^2 \Psi(w, u) - \frac{w_S}{S} \nabla^2 \Psi(u, u) - \frac{1}{S} \nabla^3 f(w_p, u_p, u_p).$$

Proof. Clearly,

$$d\Psi = dR + S^{-2}f dS - S^{-1} df.$$

Since $\nabla dR \equiv 0 \equiv \nabla dS$, this yields

$$\nabla^2\Psi = -2S^{-3}f dS \otimes dS + S^{-2} df \otimes dS + S^{-2} dS \otimes df - S^{-1}\nabla^2f. \quad (9.1.3)$$

We now use Eqs. (9.1.1) and (9.1.2) to rewrite the above as

$$\begin{aligned} \nabla^2\Psi &= -2S^{-3}g^2 dS \otimes dS + 2S^{-2}g (dg \otimes dS + dS \otimes dg) - S^{-1}(2g\nabla^2g + 2 dg \otimes dg) \\ &= -2S^{-1}(S^{-1}g dS - dg)^{\otimes 2} - 2S^{-1}g\nabla^2g. \end{aligned}$$

Taking one more derivative in Eq. (9.1.3), we obtain

$$\begin{aligned} \nabla^3\Psi(w, u, u) &= 6S^{-4}f dS(w) dS(u)^2 - 2S^{-3}df(w) dS(u)^2 - 4S^{-3}dS(w) df(u) dS(u) \\ &\quad + 2S^{-2}\nabla^2f(w, u) dS(u) + S^{-2}dS(w)\nabla^2f(u, u) - S^{-1}\nabla^3f(w, u, u) \\ &= -2S^{-1}dS(u) \nabla^2\Psi(w, u) - S^{-1}dS(w) \nabla^2\Psi(u, u) - S^{-1}\nabla^3f(w, u, u). \quad \square \end{aligned}$$

Corollary 9.1.4. *Let $D = \{(p, R, S) \in M \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} : RS - f(p) > 0\}$. Then the function $F: D \rightarrow \mathbb{R}$ defined by $F(p, R, S) = -\log(R - S^{-1}d(p, p_0)^2)$ is convex.*

9.2. Positive definite matrices

In this section, we specialize to the space $PD(n) = PD(n, \mathbb{C})$ of positive definite Hermitian $n \times n$ matrices, which is a Hadamard manifold when endowed with a well-known Riemannian metric. We collect a number of well-known results from the literature and then derive explicit formulas for the higher derivatives of the squared distance on this space by using techniques from matrix analysis. The main result of this section is Theorem 9.2.11, where we show that the squared distance is self-concordant on $PD(n)$. As explained in the introduction, this implies that the squared distance is self-concordant on arbitrary Hadamard symmetric spaces.

We will often use notation of the form $h(P)$ where $h: \mathbb{R}_{>0} \rightarrow \mathbb{R}$ is some scalar-valued function, which refers to the Hermitian matrix obtained by expanding P in an eigenbasis and applying h to its eigenvalues. Examples include but are not limited to expressions of the form P^t with $t \in \mathbb{R}$, $P + \lambda = P + \lambda I$ where $\lambda \in \mathbb{R}$, $\log(P)$, et cetera.

We think of $PD(n)$ as an open submanifold of the $n \times n$ Hermitian matrices $\text{Herm}(n) \subseteq \mathbb{C}^{n \times n}$, so that we can identify $T_P PD(n) \cong \text{Herm}(n)$ at any $P \in PD(n)$. Concretely, $X \in \text{Herm}(n)$ corresponds to the tangent vector of the curve $t \mapsto P + Xt = P^{1/2}(I + tP^{-1/2}XP^{-1/2})P^{1/2}$ at $t = 0$. These curves would be geodesics if we equipped $PD(n)$ with the Euclidean metric inherited from $\text{Herm}(n)$. Instead, we introduce the following Riemannian metric on $PD(n)$:

$$\langle X, Y \rangle_P := \text{Tr} \left[(P^{-1/2}XP^{-1/2})(P^{-1/2}YP^{-1/2}) \right] = \text{Tr} \left[P^{-1}XP^{-1}Y \right] \quad (9.2.1)$$

for $X, Y \in T_P PD(n)$. This is real-valued as the Hilbert-Schmidt inner product of two Hermitian matrices. Interestingly, $\langle \cdot, \cdot \rangle_P$ is also the *Euclidean* Hessian of the function $P \mapsto -\log \det(P)$, which is a Euclidean self-concordant barrier for $PD(n)$.

9. Self-concordance of the squared distance in non-positive curvature

It is immediate from the definition that for every $P \in \text{PD}(n)$, the bijection $Q \mapsto P^{1/2}QP^{1/2}$ is a Riemannian isometry of $\text{PD}(n)$, meaning it preserves inner products between tangent vectors. Then it also preserves the distance between any two points: for any $P, Q, Q' \in \text{PD}(n)$, we have

$$d(Q, Q') = d(P^{1/2}QP^{1/2}, P^{1/2}Q'P^{1/2}).$$

Therefore, if one is interested in properties of squared distance $f(P) = d(P, P_0)^2$, one may choose $P_0 = I$ without loss of generality. This will be convenient for our purposes.

We now give explicit formulas for the geodesics on $\text{PD}(n)$. For any $P \in \text{PD}(n)$, the exponential map at P reads

$$\text{Exp}_P(X) = P^{1/2}e^{P^{-1/2}XP^{-1/2}}P^{1/2} \quad (9.2.2)$$

and hence the geodesics through P take the form

$$P(t) = \text{Exp}_P(tX) = P^{1/2}e^{tP^{-1/2}XP^{-1/2}}P^{1/2}.$$

In particular, the geodesics through $P = I$ are of the form $\text{Exp}_I(tX) = e^{tX}$. From the description of the exponential map above it follows that $\text{Exp}_P: T_P\text{PD}(n) \rightarrow \text{PD}(n)$ is a smooth bijection for all P , with smooth inverse given by

$$\text{Exp}_P^{-1}(Q) = P^{1/2} \log(P^{-1/2}QP^{-1/2})P^{1/2}.$$

By the Hopf–Rinow theorem, there exists a length-minimizing geodesic, which is unique by the bijectivity of the exponential map; hence the distance induced by the Riemannian metric is

$$d(P, Q) = \|\log(P^{-1/2}QP^{-1/2})\|_{\text{HS}} = \|\log(Q^{-1/2}PQ^{-1/2})\|_{\text{HS}}.$$

where $\|\cdot\|_{\text{HS}}$ denotes the Hilbert–Schmidt (Frobenius) norm, because $d(P, Q) = \|\text{Exp}_P^{-1}(Q)\|_P$.

The geodesics on $\text{PD}(n)$ can be naturally described using the operator geometric mean, which is defined for $P, Q \in \text{PD}(n)$ and $t \in [0, 1]$ to be

$$P\#_t Q := P^{1/2}(P^{-1/2}QP^{-1/2})^t P^{1/2}.$$

The above formula for the geodesics through P shows that this is equal to $\text{Exp}_P(t\text{Exp}_P^{-1}(Q))$, and so it is the “time- t ”-geodesic-midpoint between P and Q .

One can also explicitly describe the parallel transport along geodesics. For $P, Q \in \text{PD}(n)$, the parallel transport of $X \in T_P\text{PD}(n)$ along the unique geodesic from P to Q is given by¹

$$\tau_{P \rightarrow Q}(X) = P^{1/2}(P^{-1/2}QP^{-1/2})^{1/2}P^{-1/2}XP^{-1/2}(P^{-1/2}QP^{-1/2})^{1/2}P^{1/2}. \quad (9.2.3)$$

¹One way of proving Eq. (9.2.3) is as follows [Sak96, Lem. IV.6.2]: for every $P \in \text{PD}(n)$, the *geodesic inversion* map $s_P: \text{PD}(n) \rightarrow \text{PD}(n)$ given by $s_P(Q) = \text{Exp}_P(-\text{Exp}_P^{-1}(Q)) = PQ^{-1}P$ is an isometry (more generally, the maps $Q \mapsto Q^{-1}$ and $Q \mapsto AQA^*$ are isometries for every $A \in \text{GL}(n, \mathbb{C})$). Let $P_0, P_1 \in \text{PD}(n)$, and let $\gamma: \mathbb{R} \rightarrow M$ be the unique geodesic such that $\gamma(0) = P_0$ and $\gamma(1) = P_1$. Then $s_{P_0}(\gamma(t)) = \gamma(-t)$ and $s_{P_1}(\gamma(t)) = \gamma(1-t)$. If X_t is a parallel vector field along γ , then so is $d(s_{P_0})(X_{-t})$, as s_{P_0} is an isometry; but $d(s_{P_0})_{P_0} = -I_{T_{P_0}\text{PD}(n)}$, and so $d(s_{P_0})(X_{-t}) = -X_t$ by the uniqueness of parallel vector fields. Similarly, $d(s_{\gamma(1/2)})(X_{1/2-t}) = -X_{1/2+t}$, and so $d(s_{\gamma(1/2)} \circ s_{P_0})(X_0) = X_1 = \tau_{P_0 \rightarrow P_1}(X_0)$. Expanding the definition of $s_{\gamma(1/2)} \circ s_{P_0}$ (also called a *transvection*), it is easy to see that its derivative is exactly the right-hand side in Eq. (9.2.3).

This may be conveniently restated as

$$\tau_{P \rightarrow \text{Exp}_P(tY)}(X) = P^{1/2}(e^{\frac{t}{2}P^{-1/2}YP^{-1/2}})P^{-1/2}XP^{-1/2}(e^{\frac{t}{2}P^{-1/2}YP^{-1/2}})P^{1/2} \quad (9.2.4)$$

which for the geodesics emanating from the identity specializes to

$$\tau_{I \rightarrow e^{tZ}}(X) = e^{\frac{t}{2}Z}Xe^{\frac{t}{2}Z},$$

i.e.,

$$\tau_{I \rightarrow Q}(X) = Q^{1/2}XQ^{1/2}.$$

Now consider a function $f: \text{PD}(n) \rightarrow \mathbb{R}$. It follows from the previous considerations and the discussion in Section 6.3 that the third derivative at $I \in \text{PD}(n)$ can be computed as follows for $X, Z \in T_I\text{PD}(n)$:

$$\begin{aligned} (\nabla^3 f)_I(Z, X, X) &= \partial_{t=0}(\nabla^2 f)_{\text{Exp}_I(tZ)}(\tau_{I \rightarrow \text{Exp}_I(tZ)}(X), \tau_{I \rightarrow \text{Exp}_I(tZ)}(X)) \\ &= \partial_{t=0}(\nabla^2 f)_{e^{tZ}}(e^{\frac{t}{2}Z}Xe^{\frac{t}{2}Z}, e^{\frac{t}{2}Z}Xe^{\frac{t}{2}Z}). \end{aligned}$$

Although we will not need it explicitly, one can also use the above to determine the covariant derivative of a general vector field. More precisely, the covariant derivative $\nabla_X Y$, where $X \in T_P\text{PD}(n)$ and $Y(t)$ is a vector field defined along the curve $P(t) = \text{Exp}_P(tX)$, is given by

$$\nabla_X Y = \partial_{t=0} \tau_{P(t) \rightarrow P}(Y(t)).$$

For $P = I$, we have

$$\nabla_X Y = \partial_{t=0} \tau_{e^{tX} \rightarrow I}(Y(t)) = \partial_{t=0} e^{-\frac{t}{2}X} Y(t) e^{-\frac{t}{2}X} = \dot{Y}(0) - \frac{1}{2} \{X, Y(0)\}$$

where we write $\{X, Y\} = XY + YX$ for the anticommutator of X and Y .

Lastly, we have an explicit expression for the Riemann curvature tensor on $\text{PD}(n)$. The fact that the curvature tensor is of this form follows from [Hel79, Thm. IV.4.2], and the prefactor of $\frac{1}{4}$ can be deduced from the fact that $\text{SPD}(2, \mathbb{C})$ is a model space for constant curvature $-\frac{1}{2}$ (the prefactor appears because we work directly with positive-definite matrices, rather than the quotient $\text{GL}_n(\mathbb{C})/\text{U}(n)$). Alternatively, one may consult the self-contained explicit proof available in [DP15]:

Lemma 9.2.1. *The Riemann curvature (1,3)-tensor at $P \in \text{PD}(n)$ is given by*

$$R(X, Y)Z = -\frac{1}{4}[[P^{-1/2}XP^{-1/2}, P^{-1/2}YP^{-1/2}], P^{-1/2}ZP^{-1/2}]$$

for every $X, Y, Z \in T_P\text{PD}(n)$. In particular, the curvature tensor is parallel along any geodesic.

This last property may be more succinctly stated as follows: if one thinks of R as a $(0, 4)$ -tensor, then $\nabla R \equiv 0$. Therefore $\text{PD}(n)$ is a *locally symmetric space*, see [Lee18, Thm. 10.19], and because it is simply connected, it is also a globally symmetric space. A simple computation using the above lemma shows that $\text{PD}(n)$ has sectional curvatures bounded by an n -independent constant with our normalization of the metric:

9. Self-concordance of the squared distance in non-positive curvature

Lemma 9.2.2. *The space $PD(n)$ has all sectional curvatures in $[-\frac{1}{2}, 0]$.*

Proof. Let $X, Y \in T_I PD(n) = \text{Herm}(n)$ have $\|X\|_I = \|Y\|_I = 1$ and $\langle X, Y \rangle_I = \text{Tr}[XY] = 0$. Assume without loss of generality that Y is diagonal. Then

$$\langle R(X, Y)Y, X \rangle = -\frac{1}{4} \sum_{i,j=1}^n |X_{ij}|^2 (Y_{jj} - Y_{ii})^2.$$

This is clearly at most 0, and

$$\sum_{i,j=1}^n |X_{ij}|^2 (Y_{jj} - Y_{ii})^2 \leq 2 \sum_{i,j=1, i \neq j}^n |X_{ij}|^2 (Y_{jj}^2 + Y_{ii}^2) \leq 2 \sum_{i,j=1}^n |X_{ij}|^2 \|Y\|_I^2 = 2 \|X\|_I^2 \|Y\|_I^2 = 2,$$

so $K(X, Y) \geq -\frac{1}{2}$. □

We now turn to the task of computing higher derivatives of the squared distance on $PD(n)$. Recall from Section 9.2 that the distance between $P, Q \in PD(n)$ is given by $d(P, Q)^2 = \|\log(P^{-1/2}QP^{-1/2})\|_{\text{HS}}^2$. To differentiate this, we use the following integral expression for the operator logarithm: for $Q \in PD(n)$, one has

$$\log(Q) = \int_0^\infty \left(\frac{1}{I + \lambda} - \frac{1}{Q + \lambda} \right) d\lambda, \quad (9.2.5)$$

where $Q + \lambda$ is shorthand for $Q + \lambda I$, and $\frac{1}{Q + \lambda} = (Q + \lambda)^{-1}$. The advantage of this expression is that it is an integral of *rational* functions of Q , which is straightforward to differentiate using the Leibniz integral rule and the following rule for differentiating matrix inverses: if $t \mapsto Q_t \in PD(n)$ is a smooth curve defined on an open interval containing 0, then

$$\partial_{t=0}(Q_t^{-1}) = -Q_0^{-1}(\partial_{t=0}Q_t)Q_0^{-1}, \quad (9.2.6)$$

as can be seen from differentiating the identity $Q_t Q_t^{-1} = I$.

We now use this integral representation to compute derivatives of the squared distance. For convenience, we consider only the squared distance to the identity $I \in PD(n)$, but this is without loss of generality; to compute the derivatives of $d(\cdot, P)^2$ for $P \in PD(n)$, one may use the fact that $Q \mapsto P^{1/2}QP^{1/2}$ is an isometry sending I to P . First, we record the formula for the first derivative.

Proposition 9.2.3. *Let $f(Q) = d(Q, I)^2 = \|\log(Q)\|_{\text{HS}}^2$. Then for $U \in T_Q PD(n)$,*

$$df_Q(U) = 2 \text{Tr}[Q^{-1} \log(Q)U] = 2 \langle Q^{1/2} \log(Q)Q^{1/2}, U \rangle_Q,$$

where $\langle \cdot, \cdot \rangle_Q$ is the Riemannian metric in $PD(n)$ defined in Eq. (9.2.1).

Proof. Let $Q_t = \text{Exp}_Q(tU)$ be the geodesic through Q in the direction U . Then by Eq. (9.2.2), we have

$$Q_t = Q^{1/2} e^{tQ^{-1/2}UQ^{-1/2}} Q^{1/2},$$

and so

$$\partial_{t=0}f(Q_t) = \partial_{t=0}\|\log(Q_t)\|_{\text{HS}}^2 = 2 \text{Tr}[\log(Q) \cdot \partial_{t=0} \log(Q_t)].$$

To evaluate $\partial_{t=0} \log(Q_t)$, we use Eq. (9.2.5) and Eq. (9.2.6) to obtain

$$\partial_{t=0} \log(Q_t) = \partial_{t=0} \int_0^\infty \left(\frac{1}{I + \lambda} - \frac{1}{Q_t + \lambda} \right) d\lambda = \int_0^\infty \frac{1}{Q + \lambda} U \frac{1}{Q + \lambda} d\lambda.$$

Therefore

$$\partial_{t=0} f(Q_t) = 2 \operatorname{Tr} \left[\log(Q) \cdot \int_0^\infty \frac{1}{Q + \lambda} U \frac{1}{Q + \lambda} d\lambda \right] = 2 \operatorname{Tr} [Q^{-1} \log(Q) \cdot U],$$

where we used cyclicity of the trace and $\int_0^\infty \frac{1}{(q+\lambda)^2} d\lambda = q^{-1}$. \square

Remark 9.2.4. In the above proof, one may also use the curve $t \mapsto Q + tU$ instead of the geodesic, because they agree in first order: it holds that $\partial_{t=0}(Q+tU) = U = \partial_{t=0} \operatorname{Exp}_Q(tU)$, and hence first derivatives of functions are not affected. However, for the second derivative, $(\nabla^2 f)_P(U, U) = \partial_{t=0}^2 f(\operatorname{Exp}_Q(tU))$ and $\partial_{t=0}^2 f(Q + tU)$ are generally distinct; a simple example is given by the function $f(P) = \operatorname{Tr}[P]$, differentiating at $Q = I$.

Remark 9.2.5. One may observe that

$$-Q^{1/2} \log(Q) Q^{1/2} = \operatorname{Exp}_Q^{-1}(I)$$

so that $df_Q(U) = -2 \langle \operatorname{Exp}_Q^{-1}(I), U \rangle_Q$, which also follows from Lemma 9.1.1.

In the next theorem, we compute the higher covariant derivatives of the squared distance. We write $\{A, B\} := AB + BA$ for the anticommutator of two matrices.

Theorem 9.2.6. Let $f(Q) = d(Q, I)^2$, and $U, W \in T_Q \operatorname{PD}(n)$. Set $\tilde{U} = Q^{-1/2} U Q^{-1/2}$ and $\tilde{W} = Q^{-1/2} W Q^{-1/2}$. Then the second derivative of f satisfies

$$\begin{aligned} (\nabla^2 f)_Q(U, U) &= \int_0^\infty d\lambda \operatorname{Tr} \left[\frac{1}{Q + \lambda} U \frac{1}{Q + \lambda} \{Q^{-1}, U\} \right] \\ &= \int_0^\infty d\lambda \operatorname{Tr} \left[\frac{1}{Q + \lambda} \tilde{U} \frac{1}{Q + \lambda} \{Q, \tilde{U}\} \right], \end{aligned}$$

and the third derivative is given by

$$\begin{aligned} (\nabla^3 f)_Q(W, U, U) &= \int_0^\infty d\lambda \operatorname{Tr} \left[\frac{1}{Q + \lambda} \tilde{U} \frac{1}{Q + \lambda} (\tilde{U} \tilde{W} Q + Q \tilde{W} \tilde{U}) \right. \\ &\quad \left. - \frac{1}{Q + \lambda} (\tilde{U} \frac{Q}{Q + \lambda} \tilde{W} + \tilde{W} \frac{Q}{Q + \lambda} \tilde{U}) \frac{1}{Q + \lambda} \{\tilde{U}, Q\} \right]. \end{aligned}$$

Proof. For the second derivative, we use the identity $(\nabla^2 f_Q)(U, U) = \partial_{t=0}^2 f(Q_t)$ where $Q_t = \operatorname{Exp}_Q(tU)$. From Proposition 9.2.3 it follows that

$$\partial_t f(Q_t) = 2 \operatorname{Tr} [Q_t^{-1} \log(Q_t) (\partial_t Q_t)].$$

As $Q_t = \operatorname{Exp}_Q(tU) = Q^{1/2} e^{tQ^{-1/2} U Q^{-1/2}} Q^{1/2}$, we have

$$\partial_t Q_t = U Q^{-1/2} e^{tQ^{-1/2} U Q^{-1/2}} Q^{1/2}, \quad \partial_{t=0}^2 Q_t = U Q^{-1} U,$$

which together with Eq. (9.2.6) leads to

$$\begin{aligned}
 & \frac{1}{2} \partial_{t=0}^2 f(Q_t) \\
 &= \text{Tr} \left[(-Q^{-1}UQ^{-1} \log(Q)U) + Q^{-1}(\partial_{t=0} \log(Q_t))U + Q^{-1} \log(Q)(\partial_{t=0}^2 Q_t) \right] \\
 &= \text{Tr} \left[Q^{-1} \int_0^\infty \frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} d\lambda U \right] \\
 &= \int_0^\infty \text{Tr} \left[\frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} U Q^{-1} \right] d\lambda.
 \end{aligned}$$

To replace the last UQ^{-1} by $\frac{1}{2}\{U, Q^{-1}\}$, note that

$$\begin{aligned}
 \text{Tr}[(Q+\lambda)^{-1}U(Q+\lambda)^{-1}UQ^{-1}] &= \text{Tr}[(Q+\lambda)^{-1}UQ^{-1}(Q+\lambda)^{-1}U] \\
 &= \text{Tr}[(Q+\lambda)^{-1}U(Q+\lambda)^{-1}Q^{-1}U]
 \end{aligned}$$

where we first used cyclicity and next that Q^{-1} and $(Q+\lambda)^{-1}$ commute. Using the definition $\tilde{U} = Q^{-1/2}UQ^{-1/2}$ yields the statement in the lemma.

We now turn to the third derivative. Let $U, W \in T_Q \text{PD}(n)$, set $Q_t = \text{Exp}_Q(tW)$ and let $U_t = \tau_{Q \rightarrow Q_t}(U)$, explicitly given in Eq. (9.2.4):

$$U_t = \tau_{Q \rightarrow Q_t}(U) = Q^{1/2}(e^{\frac{t}{2}Q^{-1/2}WQ^{-1/2}})Q^{-1/2}UQ^{-1/2}(e^{\frac{t}{2}Q^{-1/2}WQ^{-1/2}})Q^{1/2}.$$

The two basic derivatives that we need are

$$\partial_{t=0}U_t = \frac{1}{2}(WQ^{-1}U + UQ^{-1}W), \quad \partial_{t=0}Q_t = W.$$

This yields, again using Eq. (9.2.6),

$$\begin{aligned}
 (\nabla^3 f)_Q(W, U, U) &= \partial_{t=0}(\nabla^2 f)_{Q_t}(U_t, U_t) \\
 &= \partial_{t=0} \int_0^\infty \text{Tr} \left[\frac{1}{Q_t+\lambda} U_t \frac{1}{Q_t+\lambda} \{Q_t^{-1}, U_t\} \right] d\lambda \\
 &= \int_0^\infty \text{Tr} \left[-\frac{1}{Q+\lambda} W \frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} \{Q^{-1}, U\} \right] \\
 &\quad + \frac{1}{2} \text{Tr} \left[\frac{1}{Q+\lambda} (WQ^{-1}U + UQ^{-1}W) \frac{1}{Q+\lambda} \{Q^{-1}, U\} \right] \\
 &\quad + \text{Tr} \left[-\frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} W \frac{1}{Q+\lambda} \{Q^{-1}, U\} \right] \\
 &\quad + \text{Tr} \left[\frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} \{-Q^{-1}WQ^{-1}, U\} \right] \\
 &\quad + \frac{1}{2} \text{Tr} \left[\frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} \{Q^{-1}, WQ^{-1}U + UQ^{-1}W\} \right] d\lambda \\
 &= \int_0^\infty \text{Tr} \left[\frac{1}{Q+\lambda} U \frac{1}{Q+\lambda} WQ^{-1}UQ^{-1} + \frac{1}{Q+\lambda} UQ^{-1}W \frac{1}{Q+\lambda} UQ^{-1} \right. \\
 &\quad \left. - \frac{1}{Q+\lambda} (W \frac{1}{Q+\lambda} U + U \frac{1}{Q+\lambda} W) \frac{1}{Q+\lambda} \{Q^{-1}, U\} \right] d\lambda.
 \end{aligned}$$

Substituting $W = Q^{1/2}\tilde{W}Q^{1/2}$ and $U = Q^{1/2}\tilde{U}Q^{1/2}$ yields the theorem. \square

We now explicitly compute the integral expressions in Theorem 9.2.6 in terms of the entries of the matrices \tilde{U} and W . We assume without loss of generality that $Q = \text{diag}(q_1, \dots, q_n)$ by considering the expression in an eigenbasis of Q . Furthermore, we shall assume that all q_i are distinct; expressions at general Q may be obtained by taking limits, but the inequalities we will derive automatically hold for all Q by continuity. Let us start with the second derivative. Take $U \in \text{Herm}(n)$. Then for $\tilde{U} = Q^{-1/2}UQ^{-1/2}$ we have

$$\begin{aligned} (\nabla^2 f)_Q(U, U) &= \int_0^\infty d\lambda \text{Tr} \left[\frac{1}{Q + \lambda} \tilde{U} \frac{1}{Q + \lambda} \{\tilde{U}, Q\} \right] \\ &= \sum_{k,l} \int_0^\infty d\lambda \frac{1}{q_k + \lambda} \tilde{U}_{kl} \frac{1}{q_l + \lambda} \tilde{U}_{lk} (q_k + q_l) \\ &= 2 \sum_k |\tilde{U}_{kk}|^2 + \sum_{k \neq l} |\tilde{U}_{kl}|^2 \frac{(q_k + q_l) \log(q_k/q_l)}{q_k - q_l}. \end{aligned} \quad (9.2.7)$$

where we evaluated the integral using the identities

$$\int_0^\infty \frac{1}{(x + \lambda)^2} d\lambda = \frac{1}{x}, \quad \int_0^\infty \frac{1}{(x + \lambda)(y + \lambda)} d\lambda = \frac{\log(x/y)}{x - y} \quad (9.2.8)$$

for distinct $x, y > 0$. We now evaluate the third derivative in a similar manner. The only new difficulty is in performing the integration with respect to λ , for which we record the following lemma.

Lemma 9.2.7. *For distinct $x, y, z > 0$, one has*

$$\begin{aligned} &\int_0^\infty \frac{1}{(x + \lambda)(y + \lambda)(z + \lambda)} d\lambda \\ &= \frac{z(\log(x) - \log(y)) + y(\log(z) - \log(x)) + x(\log(y) - \log(z))}{(x - y)(y - z)(x - z)}. \end{aligned}$$

Proof. One can deduce from a partial fraction decomposition that

$$\frac{(x - y)(y - z)(x - z)}{(x + \lambda)(y + \lambda)(z + \lambda)} = \frac{y - z}{x + \lambda} + \frac{z - x}{y + \lambda} + \frac{x - y}{z + \lambda},$$

and the latter integrates to

$$\begin{aligned} - \int_0^\infty \frac{y - z}{x + \lambda} + \frac{z - x}{y + \lambda} + \frac{x - y}{z + \lambda} d\lambda &= \int_0^\infty (y - z) \left(\frac{1}{1 + \lambda} - \frac{1}{x + \lambda} \right) d\lambda \\ &\quad + \int_0^\infty (z - x) \left(\frac{1}{1 + \lambda} - \frac{1}{y + \lambda} \right) d\lambda \\ &\quad + \int_0^\infty (x - y) \left(\frac{1}{1 + \lambda} - \frac{1}{z + \lambda} \right) d\lambda \\ &= (y - z) \log(x) + (z - x) \log(y) + (x - y) \log(z). \quad \square \end{aligned}$$

For convenience we will use the following notation. Define $H: \mathbb{R}_{>0}^2 \rightarrow \mathbb{R}$ by

$$H(x, y) = \frac{(x + y) \log(x/y)}{x - y}, \quad (9.2.9)$$

9. Self-concordance of the squared distance in non-positive curvature

if $x, y > 0$ are distinct, and

$$H(x, x) = 2. \quad (9.2.10)$$

Next, we define $T: \mathbb{R}_{>0}^3 \rightarrow \mathbb{R}$ by

$$T(x, y, z) = \frac{x+y}{x-y} \left(\frac{x+z}{x-z} \log(x/z) - \frac{y+z}{y-z} \log(y/z) \right), \quad (9.2.11)$$

for distinct $x, y, z > 0$. Then T extends to a continuous function on $\mathbb{R}_{>0}^3$, such that

$$\begin{aligned} T(x, x, z) &= \frac{2x^2 - 2z^2 - 4xz \log(x/z)}{(x-z)^2}, \\ T(x, y, x) &= \frac{2x^2 - 2y^2 - (x+y)^2 \log(x/y)}{(x-y)^2}, \\ T(x, x, x) &= 0. \end{aligned} \quad (9.2.12)$$

Furthermore, $T(x, y, z) = T(y, x, z)$, $T(x^{-1}, y^{-1}, z^{-1}) = -T(x, y, z)$, and for every $c > 0$ we have $T(cx, cy, cz) = T(x, y, z)$.

Proposition 9.2.8. *Let $f(Q) = d(Q, I)^2$ and $U, W \in T_Q \text{PD}(n)$. Then for $Q = \text{diag}(q_1, \dots, q_n)$, and $\tilde{U} = Q^{-1/2} U Q^{-1/2}$, $\tilde{W} = Q^{-1/2} W Q^{-1/2}$, one has*

$$\begin{aligned} (\nabla^2 f)_Q(U, U) &= \sum_{k,l=1}^n |\tilde{U}_{kl}|^2 H(q_k, q_l), \\ (\nabla^3 f)_Q(W, U, U) &= \sum_{k,l,m=1}^n \tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk} T(q_k, q_l, q_m) \end{aligned} \quad (9.2.13)$$

where $H: \mathbb{R}_{>0}^2 \rightarrow \mathbb{R}$ and $T: \mathbb{R}_{>0}^3 \rightarrow \mathbb{R}$ are defined in Eqs. (9.2.9) to (9.2.12), and the subscripts refer to the respective matrix entries.

Proof. The formula for the Hessian of f was already derived in Eq. (9.2.7). For the third derivative, one can evaluate the trace in Theorem 9.2.6 as

$$\begin{aligned} \text{Tr} [\tilde{W} Q(Q + \lambda)^{-1} \tilde{U} (Q + \lambda)^{-1} \tilde{U}] &= \sum_{k,l,m} \tilde{W}_{kl} \frac{q_l}{q_l + \lambda} \tilde{U}_{lm} \frac{1}{q_m + \lambda} \tilde{U}_{mk}, \\ \text{Tr} [\tilde{W} \tilde{U} (Q + \lambda)^{-1} \tilde{U} (Q + \lambda)^{-1} Q] &= \sum_{k,l,m} \tilde{W}_{kl} \tilde{U}_{lm} \frac{1}{q_m + \lambda} \tilde{U}_{mk} \frac{q_k}{q_k + \lambda}, \end{aligned}$$

and

$$\begin{aligned} &\text{Tr} [\tilde{W} (Q + \lambda)^{-1} \{ \tilde{U}, Q \} (Q + \lambda)^{-1} \tilde{U} (Q + \lambda)^{-1} Q] \\ &= \sum_{k,l,m} \tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk} \frac{q_k (q_l + q_m)}{(q_l + \lambda)(q_m + \lambda)(q_k + \lambda)} \\ &\text{Tr} [\tilde{W} Q (Q + \lambda)^{-1} \tilde{U} (Q + \lambda)^{-1} \{ \tilde{U}, Q \} (Q + \lambda)^{-1}] \\ &= \sum_{k,l,m} \tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk} \frac{q_l (q_k + q_m)}{(q_l + \lambda)(q_m + \lambda)(q_k + \lambda)}, \end{aligned}$$

so that the third derivative satisfies

$$(\nabla^3 f)_Q(W, U, U) = \int_0^\infty d\lambda \sum_{k,l,m} \tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk} \left(\frac{q_k}{(q_k + \lambda)(q_m + \lambda)} \left(1 - \frac{q_l + q_m}{q_l + \lambda}\right) + \frac{q_l}{(q_l + \lambda)(q_m + \lambda)} \left(1 - \frac{q_k + q_m}{q_k + \lambda}\right) \right).$$

Using Eq. (9.2.8) and Lemma 9.2.7, the klm -term of this sum integrates to (interpreting expressions as limits whenever not all q_k, q_l, q_m are distinct)

$$\frac{q_k \log(q_k/q_m)}{q_k - q_m} + \frac{q_l \log(q_l/q_m)}{q_l - q_m} - \frac{(q_k(q_l + q_m) + q_l(q_k + q_m))(q_m \log(q_k/q_l) + q_l \log(q_m/q_k) + q_k \log(q_l/q_m))}{(q_k - q_l)(q_l - q_m)(q_k - q_m)},$$

which, as a short calculation reveals, is equal to

$$\frac{q_k + q_l}{q_k - q_l} \left(\frac{q_k + q_m}{q_k - q_m} \log(q_k/q_m) - \frac{q_l + q_m}{q_l - q_m} \log(q_l/q_m) \right) = T(q_k, q_l, q_m),$$

yielding the desired expression for the third derivative. \square

We note here that Proposition 9.2.8 can be used to verify that the squared distance is 2-strongly convex, which is a general property of Hadamard manifolds as mentioned before. Indeed, $\|U\|_Q = \|\tilde{U}\|_{HS}$ by definition of the Riemannian metric, so one has to show that $(\nabla^2 f)_Q(U, U) \geq 2\|\tilde{U}\|_{HS}^2$. In view of Eq. (9.2.13), it suffices to prove that $H(x, y) \geq 2$. This follows directly from the logarithmic-arithmetic mean inequality: for every $x, y > 0$, one has

$$\frac{x - y}{\log(x) - \log(y)} \leq \frac{x + y}{2}, \quad (9.2.14)$$

where the quantity $(x - y)/(\log(x) - \log(y))$ is known as the *logarithmic mean* of x and y (it is defined as x when $x = y$). It is known to be inbetween the geometric and arithmetic mean of x and y [Car72]. A short proof of Eq. (9.2.14) is as follows. Assume without loss of generality that $x < y$; then the lower bound of the Hermite–Hadamard inequality applied to the function $z \mapsto 1/z$ yields

$$\frac{\log(y) - \log(x)}{y - x} = \frac{1}{y - x} \int_x^y \frac{1}{z} dz \geq \left(\frac{x + y}{2} \right)^{-1}.$$

One can also reverse this strategy: $PD(n)$ is a Hadamard manifold, hence the squared distance is 2-strongly convex, which in turn implies the logarithmic-arithmetic mean inequality. It would be interesting to understand whether there is a more direct relation between the logarithmic-arithmetic mean inequality and the 2-strong-convexity of the squared distance, for instance via midpoint-strong-convexity considerations.

We now study the coefficients appearing in Proposition 9.2.8 to show that the squared distance is self-concordant on $PD(n)$. Let $a = \log(q_k/q_m)$ and $b = \log(q_l/q_m)$. Then

$$T(q_k, q_l, q_m) = \coth((a - b)/2) (a \coth(a/2) - b \coth(b/2)),$$

9. Self-concordance of the squared distance in non-positive curvature

whereas the square root of the product of the coefficients of $|\tilde{W}_{kl}|^2$, $|\tilde{U}_{lm}|^2$, and $|\tilde{U}_{mk}|^2$ in $\nabla^2 f$ is

$$\sqrt{H(q_k, q_l)H(q_l, q_m)H(q_k, q_m)} = \sqrt{ab(a-b) \coth(a/2) \coth(b/2) \coth((a-b)/2)}.$$

Lemma 9.2.9. *The constant $C = \sqrt{2}$ is such that for all $a, b \in \mathbb{R}$, one has*

$$\begin{aligned} & |\coth((a-b)/2) (a \coth(a/2) - b \coth(b/2))| \\ & \leq C \sqrt{ab(a-b) \coth(a/2) \coth(b/2) \coth((a-b)/2)}. \end{aligned}$$

As a consequence, for all $x, y, z > 0$, we have

$$|T(x, y, z)| \leq C \sqrt{H(x, y)H(y, z)H(x, z)}. \quad (9.2.15)$$

Remark 9.2.10. *We conjecture, based on numerical evidence, that the optimal constant in the above inequality is $C = 1/\sqrt{2}$. Let $A(x, y) = (x + y)/2$ and $G(x, y) = \sqrt{xy}$ be the arithmetic and geometric mean, respectively. The inequality for $C = 1/\sqrt{2}$ is equivalent to the following “reverse arithmetic-geometric mean inequality”: for all $a, b \in \mathbb{R}$,*

$$\frac{A(a^2 \coth(a)^2, b^2 \coth(b)^2)}{G(a^2 \coth(a)^2, b^2 \coth(b)^2)} \leq 1 + \frac{(a-b) \tanh(a-b)}{2}.$$

Proof of Lemma 9.2.9. Consider $h(x) = x \coth(x/2)$. Then h is 1-Lipschitz: its derivative is given by

$$\partial_x h(x) = \frac{\sinh(x) - x}{\cosh(x) - 1}.$$

It is clear that $|\sinh(x) - x| \leq \cosh(x) - 1$: for $x \geq 0$, the difference is $\cosh(x) - 1 - (\sinh(x) - x) = x + e^{-x} - 1$, which is convex and has zero derivative at $x = 0$, where it evaluates to 0. For $x \leq 0$, the difference is $\cosh(x) - 1 + \sinh(x) - x = e^x - x - 1 \geq 0$.

We rewrite the left- and right-hand sides of the inequality:

$$\coth((a-b)/2) (a \coth(a/2) - b \coth(b/2)) = \frac{h(a-b)(h(a) - h(b))}{a-b}$$

and

$$\sqrt{ab(a-b) \coth(a/2) \coth(b/2) \coth((a-b)/2)} = \sqrt{h(a)h(b)h(a-b)}.$$

Therefore it suffices to prove that

$$\left| \frac{h(a) - h(b)}{a-b} \right| \leq C \sqrt{\frac{h(a)h(b)}{h(a-b)}}.$$

Because h is 1-Lipschitz, the left-hand side is at most 1.

We now claim that the following lower and upper bounds on h hold: $h(x) \geq 1 + \frac{|x|}{2}$, and $h(x) \leq 2 + |x|$. The upper bound follows from h being 1-Lipschitz and $h(0) = 2$. For the lower bound, we restrict to $x \geq 0$, in which case it

suffices to prove $x \cosh(x/2) \geq (1 + x/2) \sinh(x/2)$. This is simple: we have the estimate $x \cosh(x/2) \geq 2 \sinh(x/2)$ (by a power series comparison for $x \cosh(x)$ and $\sinh(x)$), as well as $x \cosh(x/2) \geq x \sinh(x/2)$ since $\cosh(x/2) \geq \sinh(x/2)$. Therefore $x \cosh(x/2)$ is greater than their average.

We now finish up the argument: we have

$$\frac{h(a)h(b)}{h(a-b)} \geq \frac{1 + \frac{|a|+|b|}{2} + \frac{|ab|}{4}}{2 + |a| + |b|} \geq \frac{1}{2},$$

so we conclude that

$$C \sqrt{\frac{h(a)h(b)}{h(a-b)}} \geq \frac{C}{\sqrt{2}} \geq 1 \geq \frac{h(a) - h(b)}{a - b}.$$

holds for $C = \sqrt{2}$. □

This directly implies that the squared distance is self-concordant (with an n -independent constant), hence also proving Theorem 7.3.1.

Theorem 9.2.11. *Let $C \geq 0$ be such that the inequality in Lemma 9.2.9 holds. Then the function $f: \text{PD}(n) \rightarrow \mathbb{R}$ defined by $f(Q) = d(Q, I)^2$ satisfies for $Q \in \text{PD}(n)$ and $U, W \in T_Q \text{PD}(n)$ the inequality*

$$|(\nabla^3 f)_Q(W, U, U)| \leq C \sqrt{(\nabla^2 f)_Q(W, W)} (\nabla^2 f)_Q(U, U)$$

In particular, from the choice $C = \sqrt{2}$ it follows that f is 2-self-concordant.

Proof. By Eq. (9.2.15) and consecutive applications of Cauchy–Schwarz, we have

$$\begin{aligned} & |(\nabla^3 f)_Q(W, U, U)| \\ & \leq \sum_{k,l,m} |\tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk}| |T(q_k, q_l, q_m)| \\ & \leq C \sum_{k,l,m} |\tilde{W}_{kl} \tilde{U}_{lm} \tilde{U}_{mk}| \sqrt{H(q_k, q_l) H(q_l, q_m) H(q_k, q_m)} \\ & \leq C \sqrt{\sum_{k,l} |\tilde{W}_{kl}|^2 H(q_k, q_l)} \sqrt{\sum_{k,l} \left(\sum_m |\tilde{U}_{lm} \tilde{U}_{mk}| \sqrt{H(q_l, q_m) H(q_k, q_m)} \right)^2} \\ & \leq C \sqrt{\sum_{k,l} |\tilde{W}_{kl}|^2 H(q_k, q_l)} \sqrt{\sum_{k,l} \left(\sum_m |\tilde{U}_{lm}|^2 H(q_l, q_m) \right) \left(\sum_m |\tilde{U}_{mk}|^2 H(q_k, q_m) \right)} \\ & = C \sqrt{\sum_{k,l} |\tilde{W}_{kl}|^2 H(q_k, q_l)} \sqrt{\left(\sum_{l,m} |\tilde{U}_{lm}|^2 H(q_l, q_m) \right)^2} \\ & = C \sqrt{(\nabla^2 f)_Q(W, W)} (\nabla^2 f)_Q(U, U). \end{aligned} \quad \square$$

One can use this to construct a strongly self-concordant function on the open epigraph of the squared distance using Theorem 8.2.11, hence also proving Theorem 7.3.3. By imposing an additional upper bound on the value of the squared distance one can use this to construct a barrier for the epigraph, albeit with a distance-dependent barrier parameter; see Chapter 10 for similar constructions.

9.3. Constant negative curvature

In this section, we prove that the squared distance on n -dimensional hyperbolic space \mathbb{H}^n is self-concordant with a larger self-concordance parameter, and other refinements of the self-concordance estimate. We use this to construct a barrier for the epigraph of the (squared) distance in Theorem 9.3.7, which is useful for our applications in Chapter 10. Instead of dealing just with \mathbb{H}^n , we consider the *model spaces* $M_{-\kappa}^n$ with constant sectional curvature $-\kappa < 0$ (we recall that \mathbb{H}^n is M_{-1}^n). The main result of this section is the following.

Theorem 9.3.1. *Let $n \geq 2$, $\kappa > 0$, set $M = M_{-\kappa}^n$, let $p_0 \in M$, and consider $f, g: M \rightarrow \mathbb{R}$ defined by $f(p) = d(p, p_0)^2$ and $g(p) = d(p, p_0)$. One has the following estimates:*

(i) $|(\nabla^3 f)_p(w, u, u)| \leq \sqrt{\frac{8\kappa}{27}} \sqrt{(\nabla^2 f)_p(w, w)} (\nabla^2 f)_p(u, u)$, so f is $\frac{8}{\kappa}$ -self-concordant, and this constant cannot be improved.

(ii) $|(\nabla^3 f)_p(u, u, u)| \leq \sqrt{\frac{8\kappa}{27}} ((\nabla^2 f)_p(u, u))^{3/2}$, so f is $\frac{27}{2\kappa}$ -self-concordant along geodesics, and this constant cannot be improved.

(iii) $|(\nabla^3 f)_p(w, u, u)|$
 $\leq 2\zeta\sqrt{\kappa} |dg_p(w)| ((\nabla^2 f)_p(u, u) - 2dg_p(u)^2)$
 $+ 2\sqrt{\kappa} |dg_p(u)| \sqrt{(\nabla^2 f)_p(u, u) - 2dg_p(u)^2} \sqrt{(\nabla^2 f)_p(w, w) - 2dg_p(w)^2}$
 $\leq 2\zeta\sqrt{\kappa} \|w\|_p (\nabla^2 f)_p(u, u) + 2\sqrt{\kappa} \|u\|_p \sqrt{(\nabla^2 f)_p(u, u)} \sqrt{(\nabla^2 f)_p(w, w)}$,
 where $\zeta = \sup_{x \in \mathbb{R}} |\sinh(x)^{-1} - x^{-1}| \leq \frac{1}{2}$.

The fact that f is $27/(2\kappa)$ -self-concordant along geodesics was shown in [Ji07]; the optimality of this bound and the $8/\kappa$ -self-concordance are due to H. Hirai [Hir22b] (with a weaker self-concordance estimate appearing in [NW23]).

By Lemmas 6.2.3 and 8.1.2 it suffices to prove the above estimates for $M = M_{-1}^n$ and then to appropriately rescale the estimate when the curvature changes. The estimate in (iii) is a refinement of self-concordance for f (albeit with different constants), because $2\|W\|_Q^2 \leq \|W\|_{f,Q}^2$ by the 2-strong-convexity of f (and in the presence of curvature, these norms can differ by a factor that scales with the distance to the base point and the curvature). The estimate also implies that, in the terminology of Section 8.2.2, the squared distance is compatible with *every* strongly convex function, which is relevant for computing geometric means on $M_{-\kappa}^n$ as discussed in Section 10.4. The presence of the “correction terms” $-2dg_p(u)^2$ and similar for w will also be useful for proving Theorem 9.3.7, which we use later for the purpose of computing geometric medians.

Before starting with the proof of Theorem 9.3.1, we provide estimates on some single-variable functions which we use.

Lemma 9.3.2. (i) Define $\Phi: \mathbb{R} \rightarrow \mathbb{R}$ by

$$\Phi(x) := \partial_x(x \coth(x)) = \coth(x) + x - x \coth(x)^2, \quad x \neq 0, \quad (9.3.1)$$

and $\Phi(0) = 0$. Then Φ is smooth, and for $x \in \mathbb{R}_{\geq 0}$, it holds that

$$0 \leq \Phi(x) \leq \min(x, 1), \quad (9.3.2)$$

and $\lim_{x \rightarrow \infty} \Phi(x) = 1$.

(ii) It holds that

$$\zeta := \sup_{x \in \mathbb{R}_{\geq 0}} \frac{\Phi(x)}{2x \coth(x)} = \sup_{x \in \mathbb{R}} \left| \frac{1}{\sinh(x)} - \frac{1}{x} \right| < \frac{1}{2}. \quad (9.3.3)$$

We note here that numerical evaluation suggests the value of ζ is approximately 0.23536, which is slightly smaller than $\frac{1}{3\sqrt{2}} \approx 0.23570$.

Proof. We first prove (i). By $\sinh(x) = x + (1/3!)x^3 + \dots$ and $\cosh(x) = 1 + (1/2!)x^2 + \dots$, and by the identities $\cosh(x)^2 - \sinh(x)^2 = 1$, $2 \cosh(x) \sinh(x) = \sinh(2x)$, and $2 \sinh(x)^2 = \cosh(2x) - 1$, it holds that

$$\Phi(x) = \frac{\cosh(x)}{\sinh(x)} + x(1 - \coth(x)^2) = \frac{\sinh(x) \cosh(x) - x}{\sinh(x)^2} = \frac{\sinh(2x) - 2x}{\cosh(2x) - 1} = \frac{(2x)^3/3! + \dots}{(2x)^2/2! + \dots}.$$

From this, we deduce that $\Phi(x) \geq 0$ for $x \geq 0$, and

$$\lim_{x \rightarrow 0} \Phi(x) = 0 = \Phi(0).$$

Therefore Φ is continuous at 0. The above argument shows that Φ is a ratio of the analytic functions $\sinh(2x) - 2x$ and $\cosh(2x) - 1$, and the continuity at 0 shows that Φ has no singularity at 0, which is the only zero of $\cosh(2x) - 1$; hence Φ must in fact be smooth on \mathbb{R} .

We now show that $\Phi(x) \leq \min(x, 1)$ for $x \geq 0$. We have

$$\lim_{x \rightarrow 0} x \coth(x) = \lim_{x \rightarrow 0} \frac{x(1 + x^2/2! + \dots)}{x + x^3/3! + \dots} = 1.$$

By $\partial_x(x \coth x) = \Phi(x) \geq 0$ for $x \geq 0$, we have

$$x \coth(x) \geq 1.$$

This implies that Φ is nondecreasing, since

$$\partial_x \Phi(x) = \frac{2(x \coth(x) - 1)}{\sinh(x)^2} \geq 0.$$

Thus we have

$$\sup_{x \in [0, \infty)} \Phi(x) = \lim_{x \rightarrow \infty} \Phi(x) = \lim_{x \rightarrow \infty} \coth x - x/\sinh^2 x = 1.$$

Lastly, $\Phi(x) \leq x$ follows from

$$x - \Phi(x) = \coth(x)(x \coth(x) - 1) \geq 0.$$

We now prove (ii). Observe that $\lim_{x \rightarrow 0} \sinh(x)^{-1} - x^{-1} = 0$ by two applications of L'Hôpital's rule, so $\sinh(x)^{-1} - x^{-1}$ has a continuous extension to all of \mathbb{R} . A similar argument shows that $\coth(x) - x^{-1}$ can be continuously extended to $x = 0$ with value 0. For both inequalities it suffices to treat the case $x > 0$. The inequality $|\sinh(x)^{-1} - x^{-1}| \leq \frac{1}{2}$ is equivalent to

$$|x - \sinh(x)| = \sinh(x) - x \leq \frac{x \sinh(x)}{2}.$$

9. Self-concordance of the squared distance in non-positive curvature

We have equality for $x = 0$, and

$$\partial_x(\sinh(x) - x) = \cosh(x) - 1, \quad \partial_x \sinh(x) = \sinh(x) + x \cosh(x)$$

agree for $x = 0$ as well. Differentiating once more yields

$$\partial_x^2(\sinh(x) - x) = \sinh(x), \quad \partial_x^2(x \sinh(x)) = 2 \cosh(x) + x \sinh(x).$$

Clearly, $\frac{1}{2}(2 \cosh(x) + x \sinh(x)) \geq \cosh(x) \geq \sinh(x)$, and so we have proven $\zeta \leq \frac{1}{2}$. \square

Although there are several models of M_{-k}^n in which explicit computations can be performed (such as $\text{SPD}(2, \mathbb{C})$, which is $M_{-1/2}^3$), for proving Theorem 9.3.1, we take a “model-free” approach based on Jacobi fields. For a geodesic $\gamma: [0, l] \rightarrow M$, a *Jacobi field* along γ is a vector field $X = (X(t))_{t \in [0, l]}$ along γ , where $X(t) \in T_{\gamma(t)}M$ satisfies the Jacobi equation:²

$$\nabla_{\dot{\gamma}(t)} \nabla_{\dot{\gamma}(t)} X(t) + R(X(t), \dot{\gamma}(t)) \dot{\gamma}(t) = 0, \quad t \in [0, l]. \quad (9.3.4)$$

This is a linear differential equation. Therefore, the solution $X(t)$ is uniquely determined by the initial values $X(0), \nabla_{\dot{\gamma}(0)} X(0)$, or by its boundary values $X(0), X(l)$. Jacobi fields are relevant to the task of differentiating the squared distance because they arise variation fields of geodesics: the distance $d(p_0, p)$ is the minimal length of a geodesic between p_0 and p , and varying p leads to a *family* of geodesics. More precisely, one has the following classical result:

Lemma 9.3.3 (see [Sak96, p.35, 36]). *Let $\alpha: [0, l] \times (-\varepsilon, \varepsilon) \rightarrow M$ be a smooth map such that the curve $t \mapsto \alpha(t, s)$ is a geodesic for each $s \in (-\varepsilon, \varepsilon)$. Then $d\alpha(t, 0)(\frac{\partial}{\partial s})$ is a Jacobi field along geodesic $t \mapsto \alpha(t, 0)$.*

It can also be shown that every Jacobi field (along a geodesic on a compact interval) arises in this way [Lee18, Prop. 10.4], but we will not need this fact. The derivative and the Hessian of $p \mapsto f(p) = d(p, p_0)^2$ can be determined using Jacobi fields as follows.

Lemma 9.3.4 (see [Sak96, p.108–110]). *Let $p, p_0 \in M$ be distinct points, let $\gamma: [0, l] \rightarrow M$ be the unique unit-speed geodesic with $\gamma(0) = p_0, \gamma(l) = p$, and $l := g(p) = d(p, p_0)$. For $u \in T_p M$, it holds that:*

(i) $dg_p(u) = \langle \dot{\gamma}(l), u \rangle_p,$

(ii) $df_p(u) = 2l \langle \dot{\gamma}(l), u \rangle_p,$ and

(iii) $(\nabla^2 f)_p(u, u) = 2l \langle \nabla_{\dot{\gamma}(l)} X(l), u \rangle_p,$ where X is the Jacobi field along γ under the boundary condition

$$X(0) = 0, \quad X(l) = u.$$

²The meaning of $\nabla_{\dot{\gamma}(t)}$ here is slightly different from its previous meaning: instead of acting on tensor fields on an open subset of M , it acts on tensor fields along the curve γ . The two notions agree whenever $X(t)$ is locally the restriction of a vector field on M , see [Lee18, Ch. 4] for more information.

Note that (i) and (ii) are reformulations of Eq. (9.1.1), and in light of Eq. (9.1.2), (iii) is essentially a claim about $(\nabla^2 g)_p$.

We shall use the following fact about spaces of constant curvature $-\kappa$ [Sak96, Lem. II.3.3]: their Riemann curvature tensor R satisfies

$$R(X, Y)Z = -\kappa(\langle Y, Z \rangle X - \langle X, Z \rangle Y), \quad (9.3.5)$$

where we recall that $\langle \cdot, \cdot \rangle$ is the Riemannian metric. This allows one to explicitly write down the solutions of the Jacobi equation, as given in the following lemma. While this, and explicit expressions for the Hessian of the (squared) distance are well-known (see e.g. [Sak96, p. 136, p. 154] or [Lee18, Prop. 10.12, Prop. 11.3]), we provide a proof for completeness.

Lemma 9.3.5. *Let $p, p_0 \in M = \mathbb{H}^n$ with $p \neq p_0$, and let $\gamma: [0, l] \rightarrow M$ be the unit-speed geodesic from p_0 to p with $l := g(p) = d(p, p_0)$. Let $u \in T_p M$ and decompose $u = u^\top + u^\perp$ such that $u^\top = \langle u, \dot{\gamma}(l) \rangle_p \dot{\gamma}(l)$ is the part of u parallel to $\dot{\gamma}(l)$, and u^\perp orthogonal to $\dot{\gamma}$, i.e., $\langle u^\perp, \dot{\gamma}(l) \rangle_p = 0$. Then the unique Jacobi field $X(t)$ along γ with $X(0) = 0$ and $X(l) = u$ satisfies*

$$X(t) = \frac{t}{l} \tau_{\gamma, t-l} u^\top + \frac{\sinh(t)}{\sinh(l)} \tau_{\gamma, t-l} u^\perp,$$

where $\tau_{\gamma, t-l}: T_{\gamma(l)} M \rightarrow T_{\gamma(t)} M$ is the parallel transport along γ .

Proof. It is clear that $X(l) = u$ and $X(0) = 0$. Therefore it remains to check that X is a Jacobi field: we have

$$\nabla_{\dot{\gamma}(t)} X(t) = \frac{1}{l} \tau_{\gamma, t-l} u^\top + \frac{\cosh(t)}{\sinh(l)} \tau_{\gamma, t-l} u^\perp$$

and

$$\nabla_{\dot{\gamma}(t)} \nabla_{\dot{\gamma}(t)} X(t) = \frac{\sinh(t)}{\sinh(l)} \tau_{\gamma, t-l} u^\perp.$$

From Eq. (9.3.5) it follows that

$$R(X(t), \dot{\gamma}(t))\dot{\gamma}(t) = -[X(t) - \langle X(t), \dot{\gamma}(t) \rangle_{\gamma(t)} \dot{\gamma}(t)].$$

Therefore

$$\begin{aligned} \nabla_{\dot{\gamma}(t)} \nabla_{\dot{\gamma}(t)} X(t) + R(X(t), \dot{\gamma}(t))\dot{\gamma}(t) &= \frac{\sinh(t)}{\sinh(l)} \tau_{\gamma, t-l} u^\perp - X(t) + \langle X(t), \dot{\gamma}(t) \rangle_{\gamma(t)} \dot{\gamma}(t) \\ &= -\frac{t}{l} \tau_{\gamma, t-l} u^\top + \langle X(t), \dot{\gamma}(t) \rangle_{\gamma(t)} \dot{\gamma}(t) \\ &= -\frac{t}{l} \langle u, \dot{\gamma}(l) \rangle_p \dot{\gamma}(t) + \langle X(t), \dot{\gamma}(t) \rangle_{\gamma(t)} \dot{\gamma}(t) \\ &= 0, \end{aligned}$$

where the penultimate equality follows from $u^\top = \langle u, \dot{\gamma}(l) \rangle_{\gamma(l)} \dot{\gamma}(l)$ and $\tau_{\gamma, t-l} \dot{\gamma}(l) = \dot{\gamma}(t)$, and the last equality follows from $\tau_{\gamma, t-l}$ being an isometry and $\langle u, \dot{\gamma}(l) \rangle = \langle u^\top, \dot{\gamma}(l) \rangle$. \square

Using this description of the Jacobi fields leads to the following description of the Hessian, and the third covariant derivative of the squared distance.

9. Self-concordance of the squared distance in non-positive curvature

Proposition 9.3.6. *Let $p, p_0 \in M = \mathbb{H}^n$ with $p \neq p_0$ and let $\gamma: [0, l] \rightarrow M$ be the unique geodesic from p_0 to p with $l := g(p) = d(p, p_0)$. Then $f(p) = d(p, p_0)^2$ satisfies*

$$(\nabla^2 f)_p(u, u) = 2(l \coth l) \left(\langle u, u \rangle_p - \langle u, \dot{\gamma}(l) \rangle_p^2 \right) + \langle u, \dot{\gamma}(l) \rangle_p^2, \quad (9.3.6)$$

$$\begin{aligned} (\nabla^3 f)_p(w, u, u) &= 2\Phi(l) \langle w, \dot{\gamma}(l) \rangle_p \left(\langle u, u \rangle_p - \langle u, \dot{\gamma}(l) \rangle_p^2 \right) \\ &\quad + 4(l - \Phi(l)) \langle u, \dot{\gamma}(l) \rangle_p \left(\langle w, \dot{\gamma}(l) \rangle_p \langle u, \dot{\gamma}(l) \rangle_p - \langle u, w \rangle_p \right). \end{aligned} \quad (9.3.7)$$

Proof. By Lemma 9.3.5, the Jacobi field $X(t)$ along γ with $X(0) = 0$ and $X(l) = u$ satisfies

$$X(t) = \frac{t}{l} \tau_{\gamma, t-l} u^\top + \frac{\sinh(t)}{\sinh(l)} \tau_{\gamma, t-l} u^\perp$$

where $u = u^\top + u^\perp$ is a decomposition with $u^\top = \langle u, \dot{\gamma}(l) \rangle_p \dot{\gamma}(l)$ parallel and $u^\perp = u - u^\top$ orthogonal to $\dot{\gamma}(l)$, respectively. Therefore

$$\nabla_{\dot{\gamma}(l)} X(l) = \frac{1}{l} u^\top + \frac{\cosh(l)}{\sinh(l)} u^\perp = \frac{1}{l} \langle u, \dot{\gamma}(l) \rangle_p \dot{\gamma}(l) + \frac{\cosh(l)}{\sinh(l)} (u - u^\top) \quad (9.3.8)$$

Now apply Lemma 9.3.4(iii) to obtain Eq. (9.3.6).

Consider the geodesic $s \mapsto c(s) := \text{Exp}_p(sw)$. Let $\gamma_s: [0, l] \rightarrow M$ be the geodesic from p to $c(s)$ (not necessarily parametrized by the arc-length). For $s \in (-\epsilon, \epsilon)$, let $l_s := d(c(s), p_0)$ and $u_s := \tau_{c, s} u$. Applying Eq. (9.3.6) to the reparametrized geodesic $t \mapsto \gamma_s((l/l_s)t)$ ($t \in [0, l_s]$), we obtain

$$(\nabla^2 f)_{c(s)}(u_s, u_s) = 2(l_s \coth(l_s)) \langle u_s, u_s \rangle + 2(1 - l_s \coth(l_s)) (l/l_s)^2 \langle u_s, \dot{\gamma}_s(l) \rangle^2. \quad (9.3.9)$$

By Eq. (6.3.1), the covariant derivative $(\nabla^3 f)_p(w, u, u)$ is obtained by computing the s -derivative of Eq. (9.3.9) at $s = 0$. We use that

$$\partial_{s=0} l_s = \langle \dot{\gamma}(l), w \rangle, \quad \partial_{s=0} \langle u_s, u_s \rangle = 0, \quad \partial_{s=0} \langle u_s, \dot{\gamma}_s(l) \rangle = \langle u, \nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0} \rangle,$$

where the first equality follows from Lemma 9.3.4(i), and the other two follow from $X\langle Y, Z \rangle = \langle \nabla_X Y, Z \rangle + \langle Y, \nabla_X Z \rangle$ and $\nabla_{\dot{c}(s)} u_s = 0$. Hence we have

$$\begin{aligned} (\nabla^3 f)_p(w, u, u) &= 2\Phi(l) \langle \dot{\gamma}(l), w \rangle \langle u, u \rangle + 2(-\Phi(l) - 2/l + 2 \coth l) \langle \dot{\gamma}(l), w \rangle \langle u, \dot{\gamma}(l) \rangle^2 \\ &\quad + 4(1 - l \coth l) \langle u, \dot{\gamma}(l) \rangle \langle u, \nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0} \rangle \\ &= 2\Phi(l) \langle \dot{\gamma}(l), w \rangle (\langle u, u \rangle - \langle u, \dot{\gamma}(l) \rangle^2) \\ &\quad + 4(1 - l \coth l) \langle u, \dot{\gamma}(l) \rangle [\langle u, \nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0} \rangle - \langle w, \dot{\gamma}(l) \rangle \langle u, \dot{\gamma}(l) \rangle / l]. \end{aligned} \quad (9.3.10)$$

To compute $\nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0}$, consider the (smooth) map $\alpha: [0, l] \times (-\epsilon, \epsilon) = M$ given by $(t, s) \mapsto \gamma_s(t)$. Let $\frac{\partial \alpha}{\partial s}(t, s) := d\alpha_{(t,s)}(\frac{\partial}{\partial t})$ and $\frac{\partial \alpha}{\partial t}(t, s) := d\alpha_{(t,s)}(\frac{\partial}{\partial s})$. Then $\nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0} = \nabla_{\frac{\partial \alpha}{\partial s}} \frac{\partial \alpha}{\partial t}(l, 0) = \nabla_{\frac{\partial \alpha}{\partial t}} \frac{\partial \alpha}{\partial s}(l, 0)$, since $\nabla_{\frac{\partial \alpha}{\partial s}} \frac{\partial \alpha}{\partial t} = \nabla_{\frac{\partial \alpha}{\partial t}} \frac{\partial \alpha}{\partial s} + [\frac{\partial \alpha}{\partial s}, \frac{\partial \alpha}{\partial t}]$ and $[\frac{\partial \alpha}{\partial s}, \frac{\partial \alpha}{\partial t}] = d\alpha([\frac{\partial}{\partial s}, \frac{\partial}{\partial t}]) = 0$; see [Sak96, Lem. II.2.2] or [Lee18, Lem. 6.2]. By Lemma 9.3.3, $Y(t) := \frac{\partial \alpha}{\partial s}(t, 0)$ is a Jacobi field along the geodesic γ , and satisfies $Y(0) = 0$ and $Y(l) = w$. Therefore Eq. (9.3.8) yields

$$\nabla_{\dot{c}(s)} \dot{\gamma}_s(l) \Big|_{s=0} = \nabla_{\dot{\gamma}(l)} Y(l) = \frac{1}{l} \dot{\gamma}(l) \langle w, \dot{\gamma}(l) \rangle + \coth(l) (w - \dot{\gamma}(l) \langle w, \dot{\gamma}(l) \rangle).$$

By substituting this into Eq. (9.3.10), we obtain Eq. (9.3.13). \square

We are now ready to prove Theorem 9.3.1. The arguments below are due to H. Hirai [Hir22b].

Proof of Theorem 9.3.1. We first restrict to the case $\kappa = -1$. We are going to bound

$$\sigma_p(u, w) := \frac{|(\nabla^3 f)_p(w, u, u)|}{\sqrt{(\nabla^2 f)_p(w, w)(\nabla^2 f)_p(u, u)}}, \quad u, v \in T_p M \setminus \{0\}.$$

From $d(p, p_0) = l$, it holds that $\|\dot{\gamma}(l)\| = 1$. We can also assume that $\|u\|_p = \|w\|_p = 1$. Therefore, $u, v, \dot{\gamma}(l)$ can be assumed to be unit vectors in \mathbb{R}^3 , and represented in the spherical coordinate system as $\dot{\gamma}(l) = (0, 0, 1)$, $u = (\sin \theta, 0, \cos \theta)$, $w = (\sin \varphi \cos \alpha, \sin \varphi \sin \alpha, \cos \varphi)$ for $\theta, \varphi \in [0, \pi]$ and $\alpha \in [0, 2\pi]$. By Proposition 9.3.6, we have

$$(\nabla^2 f)_p(w, w) = 2 \cos^2 \varphi + 2l \coth l \sin^2 \varphi, \quad (9.3.11)$$

$$(\nabla^2 f)_p(u, u) = 2 \cos^2 \theta + 2l \coth l \sin^2 \theta, \quad (9.3.12)$$

$$(\nabla^3 f)_p(w, u, u) = 2\Phi(l) \cos \varphi \sin^2 \theta + 4(l - \Phi(l)) \cos \theta \sin \varphi \sin \theta (-\cos \alpha). \quad (9.3.13)$$

By Lemma 9.3.2(i) the quantities $\Phi(l)$, $l - \Phi(l)$, $\sin(\theta)$, and $\sin(\varphi)$ in Eq. (9.3.13) are all non-negative. Thus

$$|(\nabla^3 f)_p(w, u, u)| \leq 2\Phi(l)|\cos(\varphi)| \sin(\theta)^2 + 4(l - \Phi(l)) \sin(\varphi) \sin(\theta)|\cos(\theta)|. \quad (9.3.14)$$

For $C := l \coth(l) \geq 1$, observe that

$$\begin{aligned} \max_{\phi \in [0, \pi]} \frac{|\cos \phi|}{\sqrt{\cos^2 \phi + C \sin^2 \phi}} &= 1, & \max_{\phi \in [0, \pi]} \frac{\sin \phi}{\sqrt{\cos^2 \phi + C \sin^2 \phi}} &= \frac{1}{\sqrt{C}}, \\ \max_{\theta \in [0, \pi]} \frac{\sin \theta |\cos \theta|}{\cos^2 \theta + C \sin^2 \theta} &= \max_{\theta \in [0, \pi]} \frac{|\tan \theta|}{1 + C \tan^2 \theta} = \max_{z \in [0, \infty)} \frac{z}{1 + Cz^2} &= \frac{1}{2\sqrt{C}}. \end{aligned}$$

Therefore

$$\begin{aligned} \sigma_p(u, w) &\leq \max_{\varphi, \theta \in [0, \pi]} \frac{2\Phi(l)|\cos \varphi| \sin^2 \theta + 4(l - \Phi(l)) \sin \varphi \sin \theta |\cos \theta|}{\sqrt{2 \cos^2 \varphi + 2C \sin^2 \varphi} (2 \cos^2 \theta + 2C \sin^2 \theta)} \\ &\leq \frac{\Phi(l)}{\sqrt{2C}} + \frac{l - \Phi(l)}{\sqrt{2C}} = \frac{\tanh(l)}{\sqrt{2}} \leq \frac{1}{\sqrt{2}}. \end{aligned}$$

This shows the 8-self-concordance of f on M_{-1}^n .

We now show that this estimate is tight. Choose $\varphi = \pi/2$, $\tan^2 \theta = 1/C$, and $\alpha \in \{0, \pi\}$. From Eq. (9.3.13) we have

$$\sigma_p(u, w) = \frac{2(l - \Phi(l))|\cos(\theta)| \sin(\theta)}{\sqrt{2C}(\cos(\theta)^2 + C \sin(\theta)^2)} = \frac{l - \Phi(l)}{\sqrt{2C}} = \frac{(l - \Phi(l)) \tanh(l)}{\sqrt{2}l} = \frac{l \coth(l) - 1}{\sqrt{2}l}.$$

For $l \rightarrow \infty$, it holds that $\sigma_p(u, w) \rightarrow 1/\sqrt{2}$, and so the estimate $\sigma_p(u, w) \leq 1/\sqrt{2}$ is tight. This completes the proof of (i). Note the choice of α guarantees that we are essentially working with $u, w, \dot{\gamma}(l) \in \mathbb{R}^2$, so the argument is still valid for $n = 2$.

9. Self-concordance of the squared distance in non-positive curvature

For (ii), we consider the case of $u = w$; then $\varphi = \theta$ and $\alpha = 0$. From Eq. (9.3.13), we have

$$(\nabla^3 f)_p(u, u, u) = 2(-2l + 3\Phi(l)) \cos \theta \sin^2 \theta. \quad (9.3.15)$$

Then we have

$$\begin{aligned} \sup_{u \in T_p M} \sigma_p(u, u) &= \max_{\theta \in [0, \pi/2]} \frac{|2l - 3\Phi(l)| \tan^2 \theta}{\sqrt{2}(1 + C \tan^2 \theta)^{3/2}} = \max_{z \in [0, \infty)} \frac{|2l - 3\Phi(l)|z}{\sqrt{2}(1 + Cz)^{3/2}} \\ &= \sqrt{\frac{2}{27}} \frac{|2l - 3(\coth l + l - l \coth^2 l)|}{C} = \sqrt{\frac{2}{27}} |-3/l - \tanh l + 3 \coth l|, \end{aligned}$$

where the maximum of $z/(1 + Cz)^{3/2}$ is attained at $z = 2/C = 2(\tanh l)/l$. The supremum of the last quantity is attained at $l \rightarrow \infty$, and equals $\sqrt{2/27}$. This implies (ii), i.e., that f is $27/2$ -self-concordant along geodesics, and that this bound is tight.

Finally we show (iii). Again, we may assume $\|w\|_p = \|u\|_p = 1$, and we use the above spherical coordinates. By Eqs. (9.3.11) and (9.3.12) and Lemma 9.3.4(i), we have

$$|\sin \theta| = \sqrt{\frac{(\nabla^2 f)_p(u, u) - 2dg_p(u)^2}{2l \coth l}}, \quad |\sin \varphi| = \sqrt{\frac{(\nabla^2 f)_p(w, w) - 2dg_p(w)^2}{2l \coth l}}.$$

By substituting these into Eq. (9.3.13) and using $dg_p(u) = \cos \theta$ and $dg_p(w) = \cos \varphi$ we obtain

$$\begin{aligned} &(\nabla^3 f)_p(w, u, u) \\ &\leq \frac{\Phi(l)}{l \coth l} |dg_p(w)| ((\nabla^2 f)_p(u, u) - 2dg_p(u)^2) \\ &+ \frac{2(l - \Phi(l))}{l \coth l} |dg_p(u)| \sqrt{(\nabla^2 f)_p(w, w) - 2dg_p(w)^2} \sqrt{(\nabla^2 f)_p(u, u) - 2dg_p(u)^2} \\ &\leq 2\zeta ((\nabla^2 f)_p(u, u) - 2dg_p(u)^2) \\ &+ 2\sqrt{(\nabla^2 f)_p(w, w) - 2dg_p(w)^2} \sqrt{(\nabla^2 f)_p(u, u) - 2dg_p(u)^2}, \end{aligned}$$

where we used Lemma 9.3.2 for the second inequality. This implies (iii) for $\kappa = 1$.

Finally, the statements for $M_{-\kappa}^n$ follow from Lemmas 6.2.3 and 8.1.2. Note for part (iii) that rescaling the Riemannian metric on M_{-1}^n by a factor $1/\kappa$ yields sectional curvature κ , and rescales the distance g by a factor $1/\sqrt{\kappa}$, so to compensate one must use the prefactors $2\zeta\sqrt{\kappa}$ and $2\sqrt{\kappa}$. \square

We now use Theorem 9.3.1 to prove the following theorem, which for $\kappa = 1$ yields Theorem 7.3.4:

Theorem 9.3.7. *Let $\kappa > 0$, $M = M_{-\kappa}^n$, $p_0 \in M$, and define $f: M \rightarrow \mathbb{R}$ by $f(p) = d(p, p_0)^2$. Define an open convex set $D \subseteq M \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ by*

$$D = \{(p, R, S) \in M \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} : RS - f(p) > 0\},$$

and define a function $F: D \rightarrow \mathbb{R}$ by

$$F(p, R, S) = -\log(RS - f(p)) + \kappa f(p)$$

Then F is convex and strongly $\frac{1}{2}$ -self-concordant. Furthermore, $\lambda_{F, \frac{1}{2}}(p, R, S)^2 \leq 4 + 4\kappa f(p)$.

Proof. Recall from Corollary 9.1.4 that F is convex. Let $u = (u_p, u_R, u_S)$ and $w = (w_p, w_R, w_S)$ be tangent vectors at $(p, R, S) \in D$. Throughout the rest of this proof, we suppress the base point (p, R, S) for derivatives. Set

$$\Psi(p, R, S) = R - S^{-1}d(p, p_0)^2.$$

Instead of immediately taking F as stated, we leave the prefactor of f as a quantity $\xi > 0$ to be chosen later. The derivative of $F = -\log \Psi - \log S + \xi f$ is given by

$$dF(u) = -\frac{1}{\Psi}d\Psi(u) - \frac{u_S}{S} + \xi df(u_p).$$

Define $A_u = d\Psi(u)/\Psi$, $B_u = \sqrt{-\nabla^2\Psi(u, u)}/\Psi$, $C_u = S^{-1}u_S$ and $D_u = \sqrt{\xi \nabla^2 f(u_p, u_p)}$. We recall from Lemma 9.1.3 that Ψ is concave, so that B_u is well-defined. The Hessian of F is then given by

$$\nabla^2 F(u, u) = \underbrace{\frac{1}{\Psi^2}(d\Psi(u))^2}_{=A_u^2} - \underbrace{\frac{1}{\Psi}\nabla^2\Psi(u, u)}_{=B_u^2} + \underbrace{\frac{1}{S^2}u_S^2}_{=C_u^2} + \underbrace{\xi \nabla^2 f(u_p, u_p)}_{=D_u^2}. \quad (9.3.16)$$

For convenience we also write $B_{uw} = -\nabla^2\Psi(u, w)$. The third derivative of F is given by

$$\begin{aligned} \nabla^3 F(w, u, u) & \quad (9.3.17) \\ &= -2\frac{1}{\Psi^3}(d\Psi(w))(d\Psi(u))^2 + 2\frac{1}{\Psi^2}(d\Psi(u))(\nabla^2\Psi(w, u)) + \frac{1}{\Psi^2}d\Psi(w)(\nabla^2\Psi(u, u)) \\ & \quad - \frac{1}{\Psi}\nabla^3\Psi(w, u, u) - 2\frac{1}{S^3}w_S u_S^2 + \xi \nabla^3 f(w_p, u_p, u_p) \\ &= -2A_w A_u^2 - 2A_u B_{uw} - A_w B_u^2 - 2C_w C_u^2 - \frac{1}{\Psi}\nabla^3\Psi(w, u, u) + \xi \nabla^3 f(w_p, u_p, u_p). \end{aligned} \quad (9.3.18)$$

It is easy to see that the first four terms in Eq. (9.3.18) are bounded by a constant multiple of $\sqrt{\nabla^2 F(w, w)}\nabla^2 F(u, u)$, and similar for the last term (by α -self-concordance of f). The term $\nabla^3\Psi(w, u, u)/\Psi$ requires more effort. Recall from Lemma 9.1.3, if $g = d(p, p_0) = \sqrt{f}$, then

$$\nabla^2\Psi = -S^{-1}(2(S^{-1}g \, dS - dg)^{\otimes 2} + (\nabla^2 f - 2 \, dg \otimes dg)),$$

and the third derivative satisfies

$$\nabla^3\Psi(w, u, u) = -2S^{-1}u_S \nabla^2\Psi(w, u) - S^{-1}w_S \nabla^2\Psi(u, u) - S^{-1}\nabla^3 f(w_p, u_p, u_p).$$

Therefore

$$\begin{aligned} \nabla^3 F(w, u, u) & \\ &= -2A_w A_u^2 - 2B_{wu}(A_u + C_u) - B_u^2(A_w + C_w) - 2C_w C_u^2 + \left(\frac{1}{\Psi S} + \xi\right)\nabla^3 f \\ &= -2A_w(A_u^2 - \frac{1}{2}B_u^2) - 2B_{wu}(A_u + C_u) - 2C_w(\frac{1}{2}B_u^2 + C_u^2) + \left(\frac{1}{\Psi S} + \xi\right)\nabla^3 f. \end{aligned}$$

9. Self-concordance of the squared distance in non-positive curvature

We now use the bound from Theorem 9.3.1(iii) and the 2-strong-convexity of f :

$$\begin{aligned}
& \left| \frac{\nabla^3 f(w_p, u_p, u_p)}{S\Psi} \right| \\
& \leq \frac{|dg(w_p)| \cdot |\nabla^2 f(u_p, u_p) - 2(dg(u_p))^2|}{S\Psi} \cdot C_1 \\
& + \frac{|dg(u_p)| \cdot \sqrt{\nabla^2 f(u_p, u_p) - 2(dg(u_p))^2} \cdot \sqrt{\nabla^2 f(w_p, w_p) - 2(dg(w_p))^2}}{S\Psi} \cdot C_2 \\
& \leq \frac{1}{\sqrt{2}} \sqrt{\nabla^2 f(w_p, w_p)} B_u^2 \cdot C_1 + \frac{1}{\sqrt{2}} \sqrt{\nabla^2 f(u_p, u_p)} B_u B_w \cdot C_2
\end{aligned}$$

where $C_1 = 2\zeta\sqrt{\kappa}$ and $C_2 = 2\sqrt{\kappa}$, and $\zeta \leq \frac{1}{2}$ is defined in Lemma 9.3.2. Furthermore, f is α -self-concordant with $\alpha = 8/\kappa$ (cf. Theorem 9.3.1(i)). The triangle inequality gives

$$\begin{aligned}
& |\nabla^3 F(w, u, u)| \\
& \leq 2|A_w(A_u^2 - \frac{1}{2}B_u^2)| + 2|B_{wu}||A_u + C_u| + 2|C_w(\frac{1}{2}B_u^2 + C_u^2)| \\
& + \sqrt{\nabla^2 f(w_p, w_p)} \left(\frac{C_1}{\sqrt{2}} B_u^2 + \frac{2\xi}{\sqrt{\alpha}} \nabla^2 f(u_p, u_p) \right) + \frac{C_2}{\sqrt{2}} |B_w||B_u| \sqrt{\nabla^2 f(u_p, u_p)} \\
& = 2|A_w(A_u^2 - \frac{1}{2}B_u^2)| + 2|B_{wu}||A_u + C_u| + 2|C_w(\frac{1}{2}B_u^2 + C_u^2)| \\
& + D_w \left| \frac{C_1}{\sqrt{2\xi}} B_u^2 + \frac{2}{\sqrt{\alpha\xi}} D_u^2 \right| + |B_w B_u| \frac{C_2}{\sqrt{2\xi}} D_u \\
& \leq 2|A_w(A_u^2 - \frac{1}{2}B_u^2)| + 2|B_w||B_u|(|A_u + C_u| + \frac{C_2}{2\sqrt{2\xi}} D_u) + 2|C_w(\frac{1}{2}B_u^2 + C_u^2)| \\
& + D_w \left| \frac{C_1}{\sqrt{2\xi}} B_u^2 + \frac{2}{\sqrt{\alpha\xi}} D_u^2 \right| \\
& \leq 2\sqrt{A_w^2 + B_w^2 + C_w^2 + D_w^2} \sqrt{L},
\end{aligned}$$

where we applied $|B_{uw}| \leq |B_u||B_w|$ to get the penultimate inequality, Cauchy-Schwarz to get the last inequality, and L is defined as

$$L = (A_u^2 - \frac{1}{2}B_u^2)^2 + |B_u|^2(|A_u + C_u| + \frac{C_2}{2\sqrt{2\xi}} D_u)^2 + (\frac{1}{2}B_u^2 + C_u^2)^2 + \left| \frac{C_1}{2\sqrt{2\xi}} B_u^2 + \frac{1}{\sqrt{\alpha\xi}} D_u^2 \right|^2.$$

We now show that $L \leq 2(\nabla^2 F(u, u))^2$ for the choice $\xi = \kappa$. First, we use that $C_1 = 2\zeta\sqrt{\kappa}$, $C_2 = 2\sqrt{\kappa}$ and $\alpha = 8/\kappa$. Therefore L is

$$\begin{aligned}
L & = (A_u^2 - \frac{1}{2}B_u^2)^2 + |B_u|^2(|A_u + C_u| + \sqrt{\frac{\kappa}{2\xi}} D_u)^2 + (\frac{1}{2}B_u^2 + C_u^2)^2 + \left| \frac{\zeta\sqrt{\kappa}}{2\sqrt{2\xi}} B_u^2 + \sqrt{\frac{\kappa}{8\xi}} D_u^2 \right|^2 \\
& = A_u^4 - A_u^2 B_u^2 + \frac{1}{4} B_u^4 + |B_u|^2(|A_u + C_u|^2 + \sqrt{\frac{2\kappa}{\xi}} |A_u + C_u| D_u + \frac{\kappa}{2\xi} D_u^2) \\
& + \frac{1}{4} B_u^4 + B_u^2 C_u^2 + C_u^4 + \frac{\zeta^2 \kappa}{2\xi} B_u^4 + \frac{\zeta \kappa}{2\xi} B_u^2 D_u^2 + \frac{\kappa}{8\xi} D_u^4
\end{aligned}$$

$$\begin{aligned}
 &= A_u^4 + B_u^4 \left(\frac{1}{2} + \frac{\zeta^2 \kappa}{2\xi} \right) + C_u^4 + \frac{\kappa}{8\xi} D_u^4 \\
 &+ 2B_u^2 |A_u| |C_u| + 2B_u^2 C_u^2 + \sqrt{\frac{2\kappa}{\xi}} B_u^2 |A_u + C_u| D_u + \frac{\kappa}{2\xi} (1 + \zeta) B_u^2 D_u^2.
 \end{aligned}$$

As $\zeta \leq \frac{1}{2}$, we have $\zeta^2 \leq \frac{1}{4}$. Therefore the choice $\xi = \kappa$ ensures that

$$\begin{aligned}
 L &\leq A_u^4 + \frac{5}{8} B_u^4 + C_u^4 + \frac{1}{8} D_u^4 \\
 &+ 2B_u^2 |A_u| |C_u| + 2B_u^2 C_u^2 + \sqrt{2} B_u^2 |A_u + C_u| D_u + \frac{3}{4} B_u^2 D_u^2 \\
 &\leq A_u^4 + \frac{5}{8} B_u^4 + C_u^4 + \frac{1}{8} D_u^4 \\
 &+ \frac{1}{2} B_u^4 + 2A_u^2 C_u^2 + \frac{\sqrt{2}}{2} B_u^2 (A_u^2 + C_u^2 + 2D_u^2) + \frac{3}{4} B_u^2 D_u^2 \\
 &\leq \frac{9}{8} (\nabla^2 F(u, u))^2 \leq 2(\nabla^2 F(u, u))^2
 \end{aligned}$$

as $\nabla^2 F(u, u) = A_u^2 + B_u^2 + C_u^2 + D_u^2$. To conclude, we have shown that

$$|\nabla^3 F(w, u, u)| \leq 2\sqrt{2} \sqrt{\nabla^2 F(w, w)} \nabla^2 F(u, u)$$

and F is $\frac{1}{2}$ -self-concordant.³

We now verify the bound on the Newton decrement. For $u = (u_p, u_R, u_S) \in T_{(p,R,S)}D$ such that $\nabla^2 F(u, u) \neq 0$ and $u_p \neq 0$, we have

$$|dF(u)| = |-A_u - C_u + \xi df(u_p)| \leq \sqrt{A_u^2 + C_u^2 + D_u^2} \sqrt{1 + 1 + \frac{\xi^2 |df(u_p)|^2}{D_u^2}},$$

and

$$\frac{\xi^2 |df(u_p)|^2}{D_u^2} = \frac{\xi^2 |df(u_p)|^2}{\xi \nabla^2 f(u_p, u_p)} \leq \xi \lambda_f(p)^2 = 2\xi f(p)$$

by Corollary 9.1.2. Since we chose $\xi = \kappa$, this shows that $\lambda_{F,1/2}(p, R, S)^2 \leq 2(2 + 2\kappa f(p))$. \square

³Bounding L by $(\nabla^2 F(u, u))^2$ would lead to 1-self-concordance of F , but it is not clear whether there is a choice of $\xi > 0$ such that F is 1-self-concordant and its Newton decrement is not too adversely affected.

10. Interior-point methods for non-commutative scaling and geometric problems

In this chapter, we discuss applications of our interior-point method framework. In Section 10.1 we show that the framework can be used to solve non-commutative optimization and scaling problems. In Sections 10.2 to 10.4, we use the previously constructed barriers for the epigraph of the squared distance on Hadamard symmetric spaces and the epigraph of the distance on the model spaces for constant negative sectional curvature to the natural geometric problems of computing minimum enclosing balls, geometric medians, and Riemannian barycenters. To achieve the above, we build on the results of Section 8.2 and Chapter 9.

10.1. Non-commutative optimization and scaling problems

In this section we show that the problem of minimizing log-norm or Kempf–Ness functions, as discussed in Section 7.4, can be solved using our interior-point methods. This leads to also naturally leads to algorithms for *scaling problems*, as explained in Section 2.6.

We briefly recap the general setup for the norm minimization problem and refer to Chapter 2 or [BFG+19] for more detail. Throughout this section we let $G \subseteq GL(n, \mathbb{C})$ be a connected algebraic Lie group which is symmetric, i.e., $g^* \in G$ for every $g \in G$. We also fix $\pi: G \rightarrow GL(V)$ to be a finite-dimensional rational complex representation of G . Let $K = G \cap U(n)$, which is a maximal compact subgroup of G , and assume that V is endowed with a K -invariant inner product $\langle \cdot | \cdot \rangle$.¹ For a non-zero vector $0 \neq v \in V$, the goal is to minimize $\|\pi(g)v\|^2 = \langle v | \pi(g)^* \pi(g) v \rangle = \langle v | \pi(g^*g) v \rangle$ over $g \in G$, where we used that $\pi(g)^* = \pi(g^*)$.² Therefore, this is equivalent to minimizing $\langle v | \pi(p) v \rangle$ over $p \in M = \{g^*g : g \in G\} = G \cap PD(n) \subseteq PD(n)$. We capture this in the following definition:³

This chapter is adapted from [HNW23].

¹Following Dirac notation, we will also write $\langle v | A | w \rangle := \langle v | A w \rangle$ for vectors $v, w \in V$ and operators A on V .

²Because K acts unitarily and the Lie algebra representation $\Pi = d\pi_1$ is complex linear, one has $\Pi(X^*) = \Pi(X)^*$ for $X \in \text{Lie}(G)$. By the polar decomposition (Theorem 2.2.16) every $g \in G$ is a product $g = k \exp(H)$ with $k \in K$ and $H \in i\text{Lie}(K)$, so $\pi(g)^* = (\pi(k) \exp(\Pi(H)))^* = \exp(\Pi(H)) \pi(k)^{-1} = \exp(\Pi(H)) \pi(k^{-1}) = \pi(g^*)$ (cf. [BFG+19; Hir22a]).

³Alternatively, because of the K -invariance, $g \mapsto \|\pi(g)v\|^2$ descends to a map on the quotient $K \backslash G$. This space is naturally isometric to M via the map $Kg \mapsto g^*g$: for $G = GL(n, \mathbb{C})$ one can prove this using the polar decomposition, which generalizes to the Cartan decomposition for reductive G . As such, this is the same as Definition 10.1.1.

Definition 10.1.1. Let $M = \{g^*g : g \in G\} = G \cap \text{PD}(n) \subseteq \text{PD}(n)$. For $0 \neq v \in V$, the function $\phi_v : M \rightarrow \mathbb{R}$ is defined by

$$\phi_v : M \rightarrow \mathbb{R}, \quad \phi_v(p) = \log \langle v | \pi(p) | v \rangle. \quad (10.1.1)$$

For the special case where $G = \text{GL}(n, \mathbb{C})$, $V = \mathbb{C}^n$ and π is the identity map, we write

$$f_v : \text{PD}(n) \rightarrow \mathbb{R}, \quad f_v(P) = \log \langle v | P | v \rangle. \quad (10.1.2)$$

We note that M is a convex subset of $\text{PD}(n)$ [BH13, Thm. 10.58, Lem. 10.59], so the geodesics in M are precisely the geodesics in $\text{PD}(n)$ which lie completely in G . Thus the tangent space $T_I M$ consists of those Hermitian matrices $H \in \text{Herm}(n) = T_I \text{PD}(n)$ which also are in $\text{Lie}(G) := T_I G$, the Lie algebra of G . For $G = \text{GL}(n, \mathbb{C})$, we simply have that $T_I M = \text{Herm}(n)$.

Because K acts unitarily, π restricts to a map $M \rightarrow \text{PD}(V)$, and one can verify that it sends geodesics to geodesics (i.e., it is *geodesically affine*). At the identity, we have the explicit description

$$\pi(\text{Exp}_I(tH)) = \text{Exp}_I(t\Pi(H))$$

for $H \in T_I M$ and $\Pi : \text{Lie}(G) \rightarrow \text{End}(V) = \text{Lie}(\text{GL}(V))$ is given by the derivative of π , i.e., $\Pi = d\pi_I$. The linear map Π is also known as the Lie algebra homomorphism induced by π . Therefore, ϕ_v is the composition of the geodesically affine map $M \rightarrow \text{PD}(V)$, $p \mapsto \pi(p)$, and the map $\text{PD}(V) \rightarrow \mathbb{R}$ given by $P \mapsto \log \langle v | P | v \rangle$, i.e., the function for the defining representation of $\text{GL}(V)$. To establish bounds on the derivatives of ϕ_v , it therefore suffices to prove bounds on the derivatives of f_v , and to translate the results via Π .

Below, we prove the well-known fact that ϕ_v is convex on M (see, e.g., Proposition 2.6.6 or [BFG+19]). As explained above it suffices to prove this for the special case where $G = \text{GL}(n, \mathbb{C})$ and $V = \mathbb{C}^n$, with $\pi : G \rightarrow \text{GL}(V)$ given by the identity map.

Proposition 10.1.2. For $0 \neq v \in \mathbb{C}^n$, the Hessian of the function $f_v : \text{PD}(n) \rightarrow \mathbb{R}$ defined in Eq. (10.1.2) satisfies for every $P \in \text{PD}(n)$ and $U \in T_P \text{PD}(n)$ the identity

$$(\nabla^2 f_v)_P(U, U) = \frac{\langle \tilde{v} | (\tilde{U} - \frac{\langle \tilde{v} | \tilde{U} | \tilde{v} \rangle}{\langle \tilde{v} | \tilde{v} \rangle} I)^2 | \tilde{v} \rangle}{\langle \tilde{v} | \tilde{v} \rangle},$$

where we use the notation $\tilde{v} = P^{1/2}v$ and $\tilde{U} = P^{-1/2}UP^{-1/2}$. As a consequence, for every representation $\pi : G \rightarrow \text{GL}(V)$ and $v \in V$, ϕ_v is convex.

Proof. We compute the Hessian of $f := f_v$. First off, we have

$$\partial_t f(\text{Exp}_P(tU)) = \partial_t \log \langle v | \text{Exp}_P(tU) | v \rangle = \frac{\langle \tilde{v} | \tilde{U} e^{t\tilde{U}} | \tilde{v} \rangle}{\langle \tilde{v} | e^{t\tilde{U}} | \tilde{v} \rangle}.$$

The second derivative is given by

$$\partial_{t=0}^2 f(\text{Exp}_P(tU)) = \frac{\langle \tilde{v} | \tilde{U}^2 | \tilde{v} \rangle \langle \tilde{v} | \tilde{v} \rangle - \langle \tilde{v} | \tilde{U} | \tilde{v} \rangle^2}{\langle \tilde{v} | \tilde{v} \rangle^2} = \frac{\langle \tilde{v} | (\tilde{U} - \frac{\langle \tilde{v} | \tilde{U} | \tilde{v} \rangle}{\langle \tilde{v} | \tilde{v} \rangle} I)^2 | \tilde{v} \rangle}{\langle \tilde{v} | \tilde{v} \rangle},$$

hence is non-negative. □

The expression for the first- and second derivatives can be understood in terms of the expectation and variance of corresponding random variables, as pointed out in [BFG+19].⁴ This will be useful for bounding the third derivative. Define a linear map $\Phi_v: \mathbb{C}^{n \times n} \rightarrow \mathbb{C}$ by

$$\Phi_v(A) = \frac{\langle v|A|v \rangle}{\langle v|v \rangle}. \quad (10.1.3)$$

Then Φ_v is what is known as a *completely positive and unital* map.⁵ Such a map is to be interpreted as taking the expectation with respect to a random variable, where the random variable is now specified by a complex matrix. One can define the *covariance* between two matrices $A, B \in \mathbb{C}^{n \times n}$ as

$$\text{Cov}_v(A, B) = \Phi_v(A^*B) - \Phi_v(A)^*\Phi_v(B). \quad (10.1.4)$$

The variance of A is defined accordingly as $\text{Var}_v(A) = \text{Cov}_v(A, A)$. With this notation, we can more succinctly write

$$(\nabla^2 f_v)_P(U, U) = \text{Var}_{\tilde{v}}(\tilde{U}),$$

where $\tilde{v} = P^{1/2}v$ and $\tilde{U} = P^{-1/2}UP^{-1/2}$ as before. Then the third derivative can be computed as follows.

Proposition 10.1.3. *Let $0 \neq v \in \mathbb{C}^n$ and let $f_v: \text{PD}(n) \rightarrow \mathbb{R}$ be as defined in Eq. (10.1.2). Then for every $U, W \in \text{T}_I \text{PD}(n) = \text{Herm}(n)$, its third derivative satisfies*

$$\begin{aligned} & (\nabla^3 f_v)_I(W, U, U) \\ &= \frac{\langle v|\{W, U^2\}|v \rangle}{2 \langle v|v \rangle} - \frac{\langle v|U^2|v \rangle \langle v|W|v \rangle}{\langle v|v \rangle^2} - \frac{\langle v|U|v \rangle \langle v|\{W, U\}|v \rangle}{\langle v|v \rangle^2} + \frac{2 \langle v|U|v \rangle^2 \langle v|W|v \rangle}{\langle v|v \rangle^3} \\ &= \text{Re} \left(\text{Cov}(W, U^2 - 2\Phi(U)U) \right). \end{aligned}$$

Proof. To compute the third derivative of $f := f_v$ at $I \in \text{PD}(n)$, note that

$$\begin{aligned} & \partial_{t=0}(\nabla^2 f)_{\text{Exp}_I(tW)}(\tau_{I \rightarrow \text{Exp}_I(tW)}U, \tau_{I \rightarrow \text{Exp}_I(tW)}U) \\ &= \partial_{t=0} \left(\frac{\langle v|e^{\frac{t}{2}W}U^2e^{\frac{t}{2}W}|v \rangle}{\langle v|e^{tW}|v \rangle} - \frac{\langle v|e^{\frac{t}{2}W}Ue^{\frac{t}{2}W}|v \rangle^2}{\langle v|e^{tW}|v \rangle^2} \right) \\ &= \frac{\langle v|\{W, U^2\}|v \rangle}{2 \langle v|v \rangle} - \frac{\langle v|U^2|v \rangle \langle v|W|v \rangle}{\langle v|v \rangle^2} - \frac{\langle v|U|v \rangle \langle v|\{W, U\}|v \rangle}{\langle v|v \rangle^2} + \frac{2 \langle v|U|v \rangle^2 \langle v|W|v \rangle}{\langle v|v \rangle^3}. \end{aligned}$$

Using the map Φ and the associated covariance defined in Eqs. (10.1.3) and (10.1.4), we may rewrite the above more succinctly as

$$\begin{aligned} (\nabla^3 f)_I(W, U, U) &= \frac{1}{2} \Phi(\{W, U^2\}) - \Phi(U^2)\Phi(W) - \Phi(U)\Phi(\{W, U\}) + 2\Phi(U)^2\Phi(W) \\ &= \frac{1}{2}(\text{Cov}(W, U^2) + \text{Cov}(U^2, W)) - \Phi(U)(\text{Cov}(U, W) + \text{Cov}(W, U)) \\ &= \text{Re} \left(\text{Cov}(W, U^2 - 2\Phi(U)U) \right). \quad \square \end{aligned}$$

⁴Similarly, the higher derivatives *along geodesics* can be related to higher cumulants, see [BFG+19, Rem. 3.16].

⁵This means that $\Phi_v(I) = 1$, and the complete positivity refers to the fact that for every $n' \geq 1$, the map $\Phi_v \otimes I_{\mathbb{C}^{n' \times n'}}: \mathbb{C}^{n \times n} \otimes \mathbb{C}^{n' \times n'} \rightarrow \mathbb{C}^{n' \times n'}$ sends positive-semidefinite operators to positive-semidefinite operators.

Remark 10.1.4. The functions f_ν are not necessarily self-concordant, even along geodesics. To see this, consider $\nu = \frac{1}{\sqrt{2}}(e_1 - e_2)$ and for $z \in \mathbb{R}$ the matrix $U_z \in \text{Herm}(2)$ given by

$$U_z = \begin{bmatrix} 1 & z \\ z & 0 \end{bmatrix}.$$

Then

$$(\nabla^2 f_\nu)_I(U_z, U_z) = \frac{1}{4}, \quad (\nabla^3 f_\nu)_I(U_z, U_z, U_z) = \frac{z}{2},$$

so $|(\nabla^3 f_\nu)_I(U_z, U_z, U_z)|$ can be arbitrarily large compared to $(\nabla^2 f_\nu)_I(U_z, U_z)^{3/2}$.

Although self-concordance does not hold, we do have the following bound on its third derivative, which implies that it is compatible (in the sense of Section 8.2.2) with any strongly convex function. This generalizes [BFG+19, Prop. 3.15] beyond the case $W = U$.

Theorem 10.1.5. Let $0 \neq \nu \in \mathbb{C}^n$ and let $f_\nu: \text{PD}(n) \rightarrow \mathbb{R}$ be as defined in Eq. (10.1.2). For every $P \in \text{PD}(n)$ and $U, W \in T_P \text{PD}(n) = \text{Herm}(n)$, one has the estimate

$$\begin{aligned} |(\nabla^3 f_\nu)_P(W, U, U)| &\leq 4 \|\tilde{U}\|_\infty \sqrt{(\nabla^2 f_\nu)_P(W, W)} \sqrt{(\nabla^2 f_\nu)_P(U, U)} \\ &\leq 4 \|U\|_P \sqrt{(\nabla^2 f_\nu)_P(W, W)} \sqrt{(\nabla^2 f_\nu)_P(U, U)} \\ &= 4 \|U\|_P \|W\|_{f_\nu, P} \|U\|_{f_\nu, P}. \end{aligned}$$

where $\tilde{U} = P^{-1/2} U P^{-1/2}$, and $\|\cdot\|_\infty$ is the spectral norm.

Proof. We prove the statement for $P = I$, and set $f := f_\nu$. Writing $\text{Var}(A) = \text{Cov}(A, A)$, an operator version of the Cauchy–Schwarz inequality [BD00] yields

$$|(\nabla^3 f)_I(W, U, U)|^2 \leq |\text{Cov}(W, U^2 - 2\Phi(U)U)|^2 \leq \text{Var}(W) \text{Var}(U^2 - 2\Phi(U)U).$$

Using that for every $A, B \in \mathbb{C}^{n \times n}$,

$$\text{Var}(A + B) = \text{Var}(A) + \text{Var}(B) + \text{Cov}(A, B) + \text{Cov}(B, A) \leq 2(\text{Var}(A) + \text{Var}(B)),$$

one can deduce for Hermitian A that

$$\begin{aligned} \text{Var}(U^2 - 2\Phi(U)U) &\leq 2 \text{Var}(U(U - \Phi(U))) + 2 \text{Var}(\Phi(U)U) \\ &\leq 2 \|U^2\|_\infty \text{Var}(U - \Phi(U)) + 2\Phi(U)^2 \text{Var}(U) \\ &\leq 4 \|U\|_\infty^2 \text{Var}(U) \end{aligned}$$

where the second inequality follows from

$$\begin{aligned} \text{Var}(U(U - \Phi(U))) &\leq \Phi((U - \Phi(U))U(U - \Phi(U))) \\ &= \frac{\langle \nu | (U - \Phi(U))U(U - \Phi(U)) | \nu \rangle}{\langle \nu | \nu \rangle} \\ &\leq 2 \|U^2\|_\infty \frac{\langle \nu | (U - \Phi(U))^2 | \nu \rangle}{\langle \nu | \nu \rangle} \\ &= 2 \|U^2\|_\infty \text{Var}(U - \Phi(U)). \end{aligned}$$

The theorem now follows from the observation that $(\nabla^2 f)_I(U, U) = \text{Var}(U - \Phi(U)) = \text{Var}(U)$. \square

Corollary 10.1.6. For $0 \neq v \in V$, ϕ_v defined in Eq. (10.1.1) satisfies for all $p \in M$ and $u, w \in T_p M$ the inequality

$$|(\nabla^3 \phi_v)_p(w, u, u)| \leq 4 \|d\pi_p(u)\|_{\pi(p)} \sqrt{(\nabla^2 \phi_v)_p(w, w)} \sqrt{(\nabla^2 \phi_v)_p(u, u)}.$$

The quantity $\|d\pi_p(u)\|_{\pi(p)}$ can be understood by observing that

$$\begin{aligned} d\pi_p(u) &= \partial_{t=0} \pi(\text{Exp}_p(tu)) = \partial_{t=0} \pi \left(p^{1/2} e^{tp^{-1/2} u p^{-1/2}} p^{1/2} \right) \\ &= \pi(p^{1/2}) \Pi(p^{-1/2} u p^{-1/2}) \pi(p^{1/2}). \end{aligned}$$

Therefore $\|d\pi_p(u)\|_{\pi(p)} = \|\Pi(p^{-1/2} u p^{-1/2})\|_I$. For convenience, we write $N(\pi) = \|\Pi\|$ for the operator norm of $\Pi: \text{Lie}(G) \rightarrow \text{End}(V)$. This quantity is known as the *weight norm* of π in [BFG+19], as it is determined as the largest norm of any highest weight appearing in the decomposition of the representation π into irreducible components, see Section 2.6 for details. Then the above computation shows that the operator norm of $d_p \pi$ with respect to $\|\cdot\|_p$ and $\|\cdot\|_{\pi(p)}$ is exactly the weight norm $N(\pi)$.

Corollary 10.1.7. Let $N(\pi) = \|\Pi\|$ the weight norm of π . Then for $0 \neq v \in V$, ϕ_v defined in Eq. (10.1.1) satisfies for all $p \in M$ and $u, w \in T_p M$ the inequality

$$|(\nabla^3 \phi_v)_p(w, u, u)| \leq 4N(\pi) \|u\|_p \sqrt{(\nabla^2 \phi_v)_p(w, w)} \sqrt{(\nabla^2 \phi_v)_p(u, u)}.$$

We now apply the above to obtain an algorithmic result for optimizing ϕ_v over balls of fixed radius. Recall from Theorem 9.2.11 that $h(p) = \frac{1}{2}d(p, I)^2$ is 1-self-concordant on $\text{PD}(n)$. Therefore, the same holds on M . It directly follows from Theorem 8.2.11 that one can construct a strongly self-concordant function on its open epigraph, as h is $(0, 1)$ -compatible with itself:

Proposition 10.1.8. Let $h: M \rightarrow \mathbb{R}$ be defined by $h(p) = \frac{1}{2}d(p, I)^2$. Let $S_0 > 0$ and consider $D = \{p \in M : h(p) < S_0\}$. Then the function $F: D \rightarrow \mathbb{R}$ defined by

$$F(p) = -\log(S_0 - h(p)) + h(p)$$

is a self-concordant barrier for D with barrier parameter $\theta = 1 + S_0$.

The claim that it has barrier parameter at most $1 + S_0$ follows from $\lambda_F(p)^2 \leq 1 + \lambda_h(p)^2$ and Corollary 9.1.2. Since h is 1-strongly convex, we see that the function ϕ_v is $(0, 2N(\pi))$ -compatible with F in the sense of Definition 8.2.8. Therefore by Proposition 8.2.10, for every $t \geq 0$, the function $F_t := t\phi_v + F$ is α -self-concordant, where α is given by

$$\alpha = \begin{cases} \frac{4N(\pi)^2 - 1}{4N(\pi)^4} & \text{if } 2N(\pi)^2 > 1, \\ 1 & \text{otherwise.} \end{cases} \quad (10.1.5)$$

Lastly, we can exactly give the analytic center of F : one easily verifies that it is given by $p = I$. We obtain the following algorithmic result.

Theorem 10.1.9. For $0 \neq \nu \in V$, let $\phi_\nu: M \rightarrow \mathbb{R}$ be the function defined in Eq. (10.1.1). Let $\alpha \geq 0$ be as in Eq. (10.1.5). Then for every $S_0 > 0$, using

$$\left(\frac{9}{5} + \frac{36}{5} \sqrt{\frac{1+S_0}{\alpha}} \right) \log \left(\frac{8(1+S_0+\alpha)}{\sqrt{\alpha}\varepsilon} \right)$$

iterations of the path-following method, one can compute a point $P_\varepsilon \in M$ such that

$$\phi_\nu(P_\varepsilon) - \inf_{p \in D} \phi_\nu(p) \leq \varepsilon.$$

Proof. Set $\lambda^{(1)} = \frac{1}{4}$ and $\lambda^{(2)} = \frac{1}{9}$. Let p_i be the sequence of points defined in Theorem 8.2.17 with these choices of $\lambda^{(i)}$. These satisfy

$$\phi_\nu(p_i) - \inf_{p \in D} \phi_\nu(p) \leq \frac{2(1+S_0+\alpha) \|d\phi_\nu\|_{F,p}^*}{\sqrt{\alpha}\lambda^{(1)}} \exp \left(-i \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{(1+S_0)/\alpha}} \right)$$

where we used that the barrier parameter θ of F is $1+S_0$. The norm $\|d(\phi_\nu)_p\|_{F,p}^*$ is at most $N(\pi)$, because F is strongly 1-convex and $d(\phi_\nu)_p$ is $N(\pi)$ -Lipschitz: f_ν is easily checked to be 1-Lipschitz, and π is $N(\pi)$ -Lipschitz. Therefore we just need to ensure that

$$i \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{(1+S_0)/\alpha}} \geq \log \left(\frac{2(1+S_0+\alpha)}{\sqrt{\alpha}\lambda^{(1)}\varepsilon} \right),$$

which amounts to

$$i \geq \frac{\lambda^{(1)} + \sqrt{(1+S_0)/\alpha}}{\lambda^{(1)} - \lambda^{(2)}} \log \left(\frac{2(1+S_0+\alpha)}{\sqrt{\alpha}\lambda^{(1)}\varepsilon} \right) = \frac{9}{5} \left(1 + 4\sqrt{\frac{1+S_0}{\alpha}} \right) \log \left(\frac{8(1+S_0+\alpha)}{\sqrt{\alpha}\varepsilon} \right). \quad \square$$

Corollary 10.1.10. For $0 \neq \nu \in V$, let ϕ_ν be the function defined in Eq. (10.1.1). Then for every $\varepsilon > 0$ and $R_0 > 0$, an ε -approximate minimizer of ϕ_ν over a ball of radius R_0 around $I \in M \subseteq \text{PD}(n)$ can be found using

$$O \left((1+R_0)(1+N(\pi)) \log \left(\frac{R_0 N(\pi)}{\varepsilon} \right) \right)$$

iterations of the path-following method.

By the non-commutative duality theorem (Theorem 2.6.7), this also implies that the scaling problem (Problem 2.6.4) can be solved with precision $\varepsilon > 0$ with a complexity depending polynomially on R , $N(\pi)$, $\log(1/\varepsilon)$, and $\log(1/\gamma(\pi))$, where $\gamma(\pi)$ is the *weight margin* of the representation.

We briefly comment on the geometric meaning of the functions ϕ_ν . For the purpose of optimization, it is natural to consider whether there exists an analogue of (non-constant) linear functions on \mathbb{R}^n . This is generally not the case; in fact, if M is a complete Riemannian manifold with a non-constant smooth function $h: M \rightarrow \mathbb{R}$ such that $\nabla^2 h = 0$, then M is isometric to a product $M' \times \mathbb{R}$, such that after this identification, h is some multiple of the projection onto the

second coordinate [Inn82].⁶ There does exist another useful generalization, namely the class of Busemann functions; see [BH13, p. II.8] for general background. These may be defined on any Hadamard manifold M (and also more generally) as follows [Hir22a]: for a (not necessarily unit-speed) geodesic $\gamma: \mathbb{R} \rightarrow M$ with $\dot{\gamma} \neq 0$, define $b_\gamma: M \rightarrow \mathbb{R}$ by

$$b_\gamma(p) := \|\dot{\gamma}(0)\| \left(\lim_{t \rightarrow \infty} d(p, \gamma(t/\|\dot{\gamma}(0)\|)) - t \right). \quad (10.1.6)$$

This limit is well-defined and the resulting function turns out to be convex, and in the specific case of $M = \mathbb{R}^n$, reduces to an arbitrary (suitably normalized) affine function. For $M = \text{PD}(n)$, whenever γ converges to a *rational* point at infinity, the Busemann function is a multiple of the ϕ_v associated with a highest weight vector for an irreducible representation of $\text{GL}(n)$. This follows, e.g., by comparing [Hir22a, Lem. 2.34] and [FW20, Thm. 5.7]. The functions f_v for $v \in \mathbb{C}^n$ with $\|v\| = 1$, considered as a vector in the defining representation of $\text{GL}(n)$, correspond to those γ for which $\dot{\gamma}(0)$ is $-vv^*$, which may also be deduced from e.g. [BH13, Prop. 10.69].

10.2. The minimum enclosing ball problem

In this section we show to apply the results of Section 8.2 and Chapter 9 to various geometric problems, all of which involve the distance function or its square.

We first study the *minimum enclosing ball problem (MEB)* on a manifold M : given $m \geq 3$ distinct points p_1, p_2, \dots, p_m in M , find the smallest ball containing all of them. More formally, finding the MEB amounts to solving the following nonsmooth optimization problem:

$$\text{minimize } R \text{ s.t. } (p, R) \in M \times \mathbb{R}, \quad d(p, p_i) \leq R \quad (i = 1, 2, \dots, m). \quad (10.2.1)$$

In the case of Euclidean space $M = \mathbb{R}^n$, MEB is a well-studied problem in computational geometry, and can be formulated as a second-order cone program to which an interior-point method is applicable; see e.g. [KMY04].

Nielsen and Hadjeres [NH15] addressed this problem for a hyperbolic space M . We shall assume that M is a complete convex submanifold of $\text{PD}(n)$, but we note that similar results may be obtained for products of (rescalings of) these spaces, hence for all Hadamard symmetric spaces as explained in Section 7.3. To apply our framework, we reformulate Eq. (10.2.1) as a convex optimization problem over the following bounded domain.

Lemma 10.2.1. *Set $S_0 = \max_{i \neq j} d(p_i, p_j)^2$. Let $D \subseteq M \times \mathbb{R}$ be defined by*

$$D = \{(p, S) \in M \times \mathbb{R} \mid d(p, p_i)^2 < S < 2S_0 \quad (i = 1, 2, \dots, m)\}. \quad (10.2.2)$$

Then D is convex, open, bounded and non-empty, as $(p_j, \frac{3}{2}S_0) \in D$ for every $j = 1, \dots, m$.

⁶For Hadamard M , this may be deduced as follows: $\nabla^2 h = 0$ implies that $\|dh\|$ is a constant function on M . Since h is non-constant, $\|dh\|$ is nonzero. The gradient flow of h is by isometries, without fixed points. If $z: M \rightarrow M$ denotes the map given by following the gradient flow for time 1, then $d(z(p), p)$ is also constant as a function of $p \in M$, and the subgroup of the isometries of M generated by z acts properly by semi-simple isometries on M , in the sense of [BH13, Def. I.8.2, Def. II.6.1]. Hence by [BH13, Thm. 7.1], M splits as a product $M' \times \mathbb{R}$.

10. Interior-point methods for non-commutative scaling and geometric problems

Proof. Since D is the intersection of open epigraphs of squared distance functions and an open halfspace defined by $S < 2S_0$, it is open and convex. The boundedness of D is clear, as is the containment $(p_j, \frac{3}{2}S_0) \in D$ for every $j = 1, \dots, m$. \square

Clearly, the optimal radius of a MEB is at most $R_0 := \max_{i \neq j} d(p_i, p_j) = \sqrt{S_0}$. It is also at least half of that:

Lemma 10.2.2. *Let R_* be the optimum of Eq. (10.2.1) and $R_0 = \max_{i \neq j} d(p_i, p_j)$. Then $2R_* \geq R_0$.*

Proof. For every $p \in M$, we have

$$d(p_i, p_j) \leq d(p_i, p) + d(p, p_j) \leq 2 \max_k d(p_k, p).$$

Minimizing the right-hand side with respect to $p \in M$ yields $d(p_i, p_j) \leq 2R_*$ for every i, j ; maximizing over $i \neq j$ gives the desired bound. \square

Replacing the objective function R by $R^2 = S$, finding the MEB is equivalent to solving

$$\text{minimize } S \text{ s.t. } (p, S) \in D. \quad (10.2.3)$$

As a natural application of our results, we obtain a self-concordant barrier for D .

Proposition 10.2.3. *Let D be as in Lemma 10.2.1. Define $G: D \rightarrow \mathbb{R}$ by*

$$G(p, S) = -\log(2S_0 - S) + \sum_{i=1}^m \left(-\log(S - d(p, p_i)^2) + \frac{1}{2}d(p, p_i)^2 \right).$$

Then G is a self-concordant barrier for D , with barrier parameter $\theta = 1 + m(1 + 2S_0)$.

Proof. Let $F_i(p, S) := -\log(S - d(p, p_i)^2) + \frac{1}{2}d(p, p_i)^2$. By Corollary 7.3.2 and Theorem 7.3.3, F_i is 1-self-concordant. Furthermore, it satisfies $\lambda_{F_i}(p, S)^2 \leq 1 + d(p, p_i)^2 \leq 1 + 2S_0$. As $-\log(2S_0 - S)$ is 1-self-concordant, so is G . The Newton decrement of G then satisfies $\lambda_G(p, S)^2 \leq 1 + m(1 + 2S_0)$. Hence G is a self-concordant barrier with the claimed parameter. \square

To initialize the path-following method, we use the damped Newton method from Theorem 8.1.18. To estimate its iteration complexity, we need a lower bound on G .

Lemma 10.2.4. *For every $(p, S) \in D$, we have*

$$G(p, S) \geq -(1 + m) \log(2S_0).$$

Proof. Since $x \mapsto -\log(x)$ is decreasing, $d(p, p_i)^2 \geq 0$ and $S > 0$, we have $G(p, S) \geq -\log(2S_0) - m \log(2S_0) = -(1 + m) \log(2S_0)$. \square

The main result of this section is then the following.

Theorem 10.2.5. Let $p_1, p_2, \dots, p_m \in M$, and let R_* denote the radius of the minimum enclosing ball for these points. Set $R_0 = \max_{i \neq j} d(p_i, p_j)$. For $\varepsilon > 0$, with $O(mR_0^2)$ iterations of a damped Newton method and

$$O\left(\sqrt{1 + m(R_0^2 + 1)} \log\left(\frac{m(R_0^2 + 1)}{\varepsilon}\right)\right)$$

iterations of the path-following method, one can find $(p_\varepsilon, R_\varepsilon) \in M \times \mathbb{R}$ such that $R_\varepsilon \leq R_* + \varepsilon$, and the ball with center p_ε and radius R_ε includes p_1, p_2, \dots, p_m .

Proof. Set $\lambda^{(1)} = \frac{1}{4}$, $\lambda^{(2)} = \frac{1}{9}$. The damped Newton method of Theorem 8.1.18 with starting point $(p_j, \frac{3}{2}S_0)$ yields a point (q, S) with $\lambda_G(q, S) \leq \frac{1}{2}\lambda^{(1)}$ within the order of

$$\begin{aligned} & \frac{G(p_j, \frac{3}{2}S_0) - \inf_{(p,S) \in D} G(p, S)}{\frac{1}{2}\lambda^{(1)}} \\ & \leq \frac{-\log(S_0/2) + \sum_{i=1}^m (-\log(3S_0/2 - d(p_j, p_i)^2) + d(p_j, p_i)^2/2) + (1+m)\log(2S_0)}{\frac{1}{2}\lambda^{(1)}} \\ & \leq \frac{-\log(S_0/2) - m\log(S_0/2) + (m/2)S_0 + (1+m)\log(2S_0)}{\frac{1}{2}\lambda^{(1)}} \\ & = \frac{(1+m)\log 4 + (m/2)S_0}{\frac{1}{2}\lambda^{(1)}} \end{aligned}$$

iterations. Consider the path-following method in Theorem 8.2.17 from the initial point (q, S) , with objective $s: D \rightarrow \mathbb{R}$ defined by $(p, S) \mapsto S$. Since this is a linear map, $ts + G$ is 1-self-concordant for all $t > 0$. The starting time t_0 is given by

$$t_0 = \frac{\lambda^{(1)} - \lambda_G(q, S)}{\|ds_{(q,S)}\|_{G,(q,S)}^*} \geq \frac{\lambda^{(1)} - \lambda_G(q, S)}{2S_0},$$

where $\|ds_{(q,S)}\|_{G,(q,S)}^*$ is bounded by $2S_0$ by Lemma 8.2.18. Thus the path-following method yields a sequence of points (q_l, S_l) such that

$$S_l - R_*^2 \leq \frac{8S_0(\theta + 1)}{\lambda^{(1)}} \exp\left(-l \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{\theta}}\right),$$

where $\theta = 1 + m(1 + 2S_0)$ is the barrier parameter of G and we used $\lambda(q, S) \leq \lambda^{(1)}/2$. For $\varepsilon' > 0$, after

$$l \geq \frac{\frac{1}{4} + \sqrt{\theta}}{\frac{1}{4} - \frac{1}{9}} \log\left(\frac{32(\theta + 1)}{\varepsilon'}\right)$$

iterations, we have

$$S_l - R_*^2 \leq \varepsilon' S_0.$$

For $R_l = \sqrt{S_l}$, it holds that

$$R_l - R_* \leq \varepsilon' S_0 / (R_l + R_*) \leq \varepsilon' S_0 / 2R_* \leq \varepsilon' S_0 / R_0,$$

where the last inequality follows from Lemma 10.2.2. Therefore, choosing $\varepsilon' = \varepsilon R_0 / S_0 = \varepsilon / R_0$ yields the desired estimate. \square

10.3. The geometric median on model spaces

In this section we show how to apply the methods from Section 8.2 to compute geometric medians on the model spaces $M_{-\kappa}^n$ for constant sectional curvature $-\kappa$, where $\kappa > 0$. For now, we shall work with general M ; later, we restrict to the model spaces because it is there that we have a barrier for the epigraph of the distance function (cf. Theorem 9.3.7). Recall from the introduction that the geometric median problem is as follows: given $m \geq 3$ points $p_1, \dots, p_m \in M$, not all contained in a single geodesic, find $p_0 \in M$ such that

$$p_0 \in \operatorname{argmin}_{p \in M} s(p) := \sum_{i=1}^m d(p, p_i). \quad (10.3.1)$$

This is a convex optimization objective, as the distance to a point is convex by Lemma 9.1.1. Let us first construct define a suitable domain to optimize over.

Lemma 10.3.1. *Set $R_0 = \max_{i \neq j} d(p_i, p_j)$. Let $D \subseteq M \times \mathbb{R}$ be defined by*

$$D = \{(p, R) \in M \times \mathbb{R}^m : R_i^2 > d(p, p_i)^2, 2R_0 > R_i > 0\}.$$

Then D is convex, open, and non-empty: for every $j \in [m]$, we have $(p_j, \frac{3}{2}R_0 \mathbf{1}) \in D$, where $\mathbf{1} \in \mathbb{R}^m$ is the all-ones vector.

Proof. The convexity of D follows from the convexity of the distance function, see Lemma 9.1.1. The fact that D is open is obvious. Lastly, the given points are in D because

$$d(p_j, p_i) \leq R_0 < \frac{3}{2}R_0. \quad \square$$

Lemma 10.3.2. *Define $c: M \times \mathbb{R}^m \rightarrow \mathbb{R}$ by $c(p, R) = \sum_{i=1}^m R_i$, and let $s: M \rightarrow \mathbb{R}$ be as in Eq. (10.3.1). Then*

$$\inf_{(p, R) \in D} c(p, R) = \inf_{p \in M} s(p)$$

The proof relies on the fact that the geometric median of p_1, \dots, p_m is contained in the convex hull of these points, for which we essentially follow the argument given in [Yan10, Prop. 2.4], where this fact is proven for more general distributions (rather than just discrete distributions).

Proof. First, we observe that for fixed $(p, R) \in D$,

$$\inf_{R': (p, R') \in D} c(p, R') = \sum_{i=1}^m d(p, p_i) = s(p).$$

Thus it suffices to prove that if $p_0 \in \operatorname{argmin}_{p \in M} s(p)$, then there exists some $R \in \mathbb{R}^m$ such that $(p_0, R) \in D$. We claim that any such p_0 is in the convex hull of the p_j . From this claim one immediately deduces that $(p_0, R) \in D$ for $R = \frac{3}{2}R_0 \mathbf{1}$, since by Lemma 10.3.1, D is convex and $(p_j, \frac{3}{2}R_0 \mathbf{1}) \in D$ for every $j \in [m]$.

We now establish the claim by proving its contrapositive. Suppose p is not in the convex hull C of the points p_1, \dots, p_m , and let q be the projection of p onto C , which is automatically distinct from p . We use the notion of Alexandrov angle, which

for three points $a, b, c \in M$ with $a \neq b, c$ is defined as the unique $\angle_a(b, c) \in [0, \pi]$ such that

$$\cos \angle_a(b, c) = \frac{\langle \text{Exp}_a^{-1}(b), \text{Exp}_a^{-1}(c) \rangle_a}{d(a, b) d(a, c)}.$$

Suppose first that $q = p_j$ for some $j \in [m]$. Then $\angle_p(q, p_j) = 0$. On the other hand, if $q \neq p_j$, by [BH13, Prop. II.2.4], we have $\angle_q(p, p_j) \geq \pi/2$. On a Hadamard manifold, the angles of a triangle add to at most π , hence $\angle_p(q, p_j) \leq \pi/2$. Since we have $m \geq 3$, there must exist at least two j such that $q \neq p_j$. Furthermore, for at least one such j , the inequality must be strict: if the inequality is not strict then we must have $\angle_{p_j}(q, p) = 0$, so p, q, p_j all lie on a single geodesic. Since p is distinct from all p_j , s is differentiable at p , and it follows from Lemma 9.1.1 that

$$\text{grad}(s)_p = - \sum_{j=1}^m \frac{\text{Exp}_p^{-1}(p_j)}{d(p, p_j)}.$$

Since we have shown that $\angle_p(q, p_j) \leq \pi/2$ for every $j \in [m]$, with strict inequality for at least one j , we have

$$\langle \text{grad}(s)_p, \text{Exp}_p^{-1}(q) \rangle_p = -d(p, q) \sum_{j=1}^m \cos \angle_p(q, p_j) < 0$$

because $d(p, q) \neq 0$. In particular, $\text{grad}(s)_p \neq 0$ and p is not a minimizer of s . \square

We now construct a barrier for the domain D . From here onwards, we assume that $M = M_{-\kappa}^n$ with $\kappa > 0$.

Proposition 10.3.3. *Let D be as in Lemma 10.3.1. Define $G: D \rightarrow \mathbb{R}$ by*

$$G(p, R) = \sum_{i=1}^m \left(-\log(2R_0 - R_i) - 2 \log(R_i^2 - d(p, p_i)^2) + 2\kappa d(p, p_i)^2 \right).$$

Then G is a self-concordant barrier for D , with barrier parameter $\theta = 5m + 16m\kappa R_0^2$.

Proof. Let $\Psi(r) = -\log(2R_0 - r)$ and recall from Theorem 9.3.7 that $F_i(p, R, S) = -\log(RS - d(p, p_i)^2) + \kappa d(p, p_i)^2$ is strongly $\frac{1}{2}$ -self-concordant. Using Lemma 8.1.2 and the strong 1-self-concordance of $-\log(2R_0 - R)$, we deduce that G is strongly 1-self-concordant. Then for every $(p, R) \in D$, we have

$$d((p, R), (p_1, R_0))^2 = d(p, p_1)^2 + |R - R_0|^2 \leq 2R^2 - 2R_0R + R_0^2 \leq 8R_0^2 - 4R_0^2 + R_0^2 = 5R_0^2.$$

where d on the left-hand side refers to the distance on $M \times \mathbb{R}$. Furthermore, for every $(p, R) \in D$, the bound on $\lambda_{F_i, 1/2}(p, R, S) = \lambda_{2F_i, 1}(p, R, S)$ from Theorem 9.3.7 implies that

$$\lambda_G(p, R)^2 \leq \sum_{i=1}^m \lambda_\Psi(R_i)^2 + \lambda_{2F_i, 1}^2(p, R, R) \leq m + \sum_{i=1}^m (4 + 4\kappa d(p, p_i)^2) \leq 5m + 16m\kappa R_0^2.$$

Therefore G is a self-concordant barrier with barrier parameter $\theta = 5m + 16m\kappa R_0^2$. \square

We now consider how to initialize the path-following method for the objective

$$c(p, R) = \sum_{i=1}^m R_i,$$

which is such that $tc + G$ is 1-self-concordant for every $t \geq 0$, because c is linear. To apply Theorem 8.2.17, we need to find a point $(q, S) \in D$ such that $\lambda_G(q, S) < \lambda^{(1)}$.⁷ We can do this using the damped Newton method from Theorem 8.1.18. To bound the number of iterations, we must bound the potential gap of G .

Lemma 10.3.4. *For every $(p, R) \in D$, we have*

$$G(p, R) \geq -m \log(32R_0^5).$$

Proof. The function $x \mapsto -\log(x)$ is decreasing. Because $R_i > 0$ for every $i \in [m]$, we have $-\log(2R_0 - R_i) \geq -\log(2R_0)$. Similarly, because $R_i < 2R_0$ and $d(p, p_i) \geq 0$ for every $i \in [m]$, each $-\log(R_i^2 - d(p, p_i)^2)$ term is at least $-\log(4R_0^2)$. Hence $G(p, R) \geq -m \log(2R_0) - 2m \log(4R_0^2) = -m \log(32R_0^5)$, concluding the proof. \square

We now prove the main result of this section.

Theorem 10.3.5. *Let $p_1, \dots, p_m \in M_{-\kappa}^n$ with $\kappa > 0$ be $m \geq 3$ points, not all on one geodesic, and set $R_0 = \max_{i \neq j} d(p_i, p_j)$. Define $s(p) = \sum_{j=1}^m d(p, p_j)$, and let $\varepsilon > 0$. Then with $O((m+1)\kappa R_0^2)$ iterations of a damped Newton method and*

$$O\left(\sqrt{m(\kappa R_0^2 + 1)} \log\left(\frac{mR_0(\kappa R_0^2 + 1)}{\varepsilon}\right)\right)$$

iterations of the path-following method, one can find $p_\varepsilon \in M_{-\kappa}^n$ such that

$$s(p_\varepsilon) - \inf_{q \in M} s(q) \leq \varepsilon.$$

Proof. Set $\lambda^{(1)} = \frac{1}{4}$, $\lambda^{(2)} = \frac{1}{9}$. The damped Newton method of Theorem 8.1.18 with starting point $(p_j, \frac{3}{2}R_0\mathbf{1})$ yields a point (q, S) with $\lambda_G(q, S) \leq \frac{1}{2}\lambda^{(1)}$ within the order of

$$\begin{aligned} & \frac{G(p_j, \frac{3}{2}R_0\mathbf{1}) - \inf_{(p,R) \in D} G(p, R)}{\frac{1}{2}\lambda^{(1)}} \\ & \leq \frac{G(p_j, \frac{3}{2}R_0\mathbf{1}) + m \log(32R_0^5)}{\frac{1}{2}\lambda^{(1)}} \\ & = \frac{-m \log(\frac{R_0}{2}) - 2 \sum_{i=1}^m \log(\frac{9}{4}R_0^2 - d(p_j, p_i)^2) + m \log(32R_0^5) + 2\kappa \sum_{i=1}^m d(p_j, p_i)^2}{\frac{1}{2}\lambda^{(1)}} \\ & \leq \frac{-m \log(\frac{R_0}{2}) - 2 \sum_{i=1}^m \log(\frac{5}{4}R_0^2) + m \log(32R_0^5) + 8\kappa m R_0^2}{\frac{1}{2}\lambda^{(1)}} \end{aligned}$$

⁷For fixed q , it is easy to determine the optimal S , by explicitly solving the first-order optimality conditions.

$$= \frac{m \log\left(\frac{1024}{25}\right) + 8m\kappa R_0^2}{\frac{1}{2}\lambda^{(1)}}$$

iterations. A suitable choice of starting time for the path-following method from Theorem 8.2.17 is then

$$t_0 = \frac{\lambda^{(1)} - \lambda_G(q, S)}{\|dc_{(q,S)}\|_{G,(q,S)}^*}.$$

It remains to be shown that this is not too small. We give an upper bound on $\|dc_{(q,S)}\|_{G,(q,S)}^*$. The domain D is constructed so that $c(p, R) \leq 2mR_0$ for every $(p, R) \in D$, and $c(q, S) \geq 0$. It follows by Lemma 8.2.18 that

$$\|dc_{(q,S)}\|_{G,(q,S)}^* \leq 2mR_0,$$

and so $t_0 \geq (\lambda^{(1)} - \lambda_G(q, S))/(2mR_0)$. Therefore, initializing the algorithm from Theorem 8.2.17 with initial point (q, S) and the above t_0 yields a sequence of points (q_l, S_l) such that

$$c(q_l, S_l) - \inf_{(p,R) \in D} c(p, R) \leq \frac{4mR_0(\theta + 1)}{\lambda^{(1)}} \exp\left(-l \frac{\lambda^{(1)} - \lambda^{(2)}}{\lambda^{(1)} + \sqrt{\theta}}\right)$$

where θ is the barrier parameter of G , and we used that $\lambda^{(1)} - \lambda_G(q, S)$ is at least $\frac{1}{2}\lambda^{(1)}$. Rewriting the above and using Lemma 10.3.2 shows that

$$s(q_l) - \inf_{q \in M} s(q) \leq c(q_l, S_l) - \inf_{(q,R) \in D} c(q, R) \leq \varepsilon$$

whenever

$$l \geq \frac{\frac{1}{4} + \sqrt{\theta}}{\frac{1}{4} - \frac{1}{9}} \log\left(\frac{4mR_0(\theta + 1)}{\varepsilon}\right).$$

The theorem now follows from filling in $\theta = 5m + 16m\kappa R_0^2$. □

10.4. The Riemannian barycenter

We end this section by briefly commenting on the problem of finding the Riemannian barycenter, first introduced by Cartan, and sometimes also called the Fréchet or Karcher mean, see e.g. [Afs11] for some historical context on this topic. It is defined as follows: given points $p_1, \dots, p_m \in M$, find $p_0 \in M$

$$p_0 \in \operatorname{argmin}_{p \in M} f(p) := \sum_{i=1}^m d(p, p_i)^2.$$

The point p_0 is known as the barycenter of p_1, \dots, p_m , and is unique on Hadamard manifolds by strong convexity of f . It is trivial to find p_0 when $M = \mathbb{R}^n$ is Euclidean space, as it is given by $p_0 = \frac{1}{m} \sum_{i=1}^m p_i$. Furthermore, the solution is unique on any Hadamard manifold, as the squared distance is 2-strongly convex, and hence f is 2m-strongly convex. Even for hyperbolic space it is not clear whether one can give a closed-form solution to the above problem. However, if M has sectional curvatures

in $[-\kappa, 0]$, then f is $O(m\sqrt{\kappa}R/\tanh(R\sqrt{\kappa}))$ -smooth at p with $R = \max_j d(p, p_j)$, which follows from standard variational arguments [Lee18, Prop. 10.12, Thm. 10.22], hence the function f is well-conditioned. Therefore a standard gradient descent method gives an algorithm which converges relatively quickly; one can find an ε -approximate minimizer of f in $O(\sqrt{\kappa}R_0 \log([f(p) - \inf_q f(q)]/\varepsilon)/\tanh(\sqrt{\kappa}R_0))$ iterations, where R_0 is some a priori bound on size of the domain one restricts to, and p is the starting point. This can be deduced easily from Proposition 6.5.4, which gives guarantees for gradient descent for well-conditioned convex functions. We note that one could also apply more sophisticated first-order methods such as accelerated gradient descent to this problem, see [AS20].

It is natural to determine what complexity our interior-point methods give for this problem. In the setting of $M = M_{-\kappa}^n$, we can (up to logarithmic factors) recover the above iteration complexity. We restrict the above optimization problem to a ball of radius $R_1 = \max_{j \neq 1} d(p_1, p_j)$ around the point p_1 , and use the barrier $F(p) = -\log(R_1^2 - d(p, p_1)^2) + \kappa d(p, p_1)^2$, which has barrier parameter $1 + O(\kappa R_1^2)$. Then, observe that by Theorem 9.3.1(iii) and Lemma 8.2.9, f is $(\sqrt{2}\zeta\sqrt{\kappa}, \sqrt{2\kappa})$ -compatible with any squared distance function, as each of the $d(p, p_i)^2$'s is. As a consequence, f is $(\sqrt{2}\zeta, \sqrt{2})$ -compatible with F , and $tf + F$ is $O(1)$ -self-concordant for every $t \geq 1$ by Proposition 8.2.10. The path-following method, initialized with starting point p_1 (which is the analytic center of F), then yields an ε -approximate minimizer of f within $O((1 + \sqrt{\kappa}R_1) \log(m\kappa R_1/\varepsilon))$ iterations. While this specific choice of barrier may seem odd, it has the advantage that we know its analytic center to be p_1 , so it is easy to initialize the path-following method. This shows again that it is useful to have a general path-following method capable of dealing with compatible objectives, rather than just linear ones: if one included a barrier term for the epigraph of every $d(p, p_i)^2$, then it would both be harder to find the analytic center (for initialization), and the barrier parameter would scale with m . We note that a similar approach works on $PD(n)$ if one suitably generalized Theorem 9.3.1(iii).

Part III.

Quantum algorithms and lower bounds for scaling

11. An introduction to quantum algorithms and lower bounds

The purpose of this chapter is to provide a brief introduction to quantum algorithms and quantum lower bounds. We also collect some well-known results that are used in the later chapters. We shall generally be brief, and defer to the many excellent sources on quantum computing for more structured discussions of this topic, among which are [NC02; Wol22; Chi22].

11.1. Quantum computing

To understand the most common (theoretical) model for quantum computation, it is useful to contrast it with a similar classical model of computation. Although the typical formal definition of a classical computer uses Turing machines, generalizing this to a quantum setting is not as straightforward or easy to work with [Deu85; Deu89]; one particular issue is the unphysical nature of a tape head whose location is in superposition.

Instead, one can use a different classical starting point. A classical computer can be alternatively described as a device which has a memory, represented by some bit string $x \in \{0, 1\}^N$, and which performs certain basic operations in succession. A standard choice of basic operations consists of the AND-, OR- and NOT-gates: these can be specified to take their inputs from the memory x at locations i, j (or a single index in the case of NOT), and to write their output to some location i . The final output of the computation is then a bit (or sequence of bits) stored in a prespecified location in the memory.

To go from the above classical model to a quantum model of computation, one can then proceed as follows [Deu89; Yao93]. First, the memory $x \in \{0, 1\}^N$ is replaced by an N -qubit *pure state* $|\psi\rangle \in (\mathbb{C}^2)^{\otimes N}$, which is a unit vector with respect to the usual Hilbert–Schmidt inner product, i.e., $\langle\psi|\psi\rangle = \|\psi\|^2 = 1$. Such a state can be viewed as a *superposition* of the *standard basis states* $|x\rangle \in (\mathbb{C}^2)^{\otimes N} \cong \mathbb{C}^{2^N}$ labelled by $x \in \{0, 1\}^N$.

Next, one has to find a suitable extension of the basic gates. The linearity of the Schrödinger equation, which governs the time-evolution of a quantum state subject to some Hamiltonian, suggests that if $U: (\mathbb{C}^2)^{\otimes N} \rightarrow (\mathbb{C}^2)^{\otimes N}$ is a basic operation, then U should be *linear*. Since an operation should also leave the memory in a pure state again, the basic operations necessarily have to be unitary, i.e., $UU^* = U^*U = I$. Consequently, quantum operations are necessarily *reversible*. Furthermore, every *reversible* classical gate yields a permutation of the basis states, and hence can be extended to a unitary operation. In particular, the AND- and OR-gates do not

This chapter is partially adapted from [AGL+21; GN22; AGN23].

directly generalize to the quantum setting, since they are not bijections (as they are maps $\{0, 1\}^2 \rightarrow \{0, 1\}$).

A standard choice of universal gate set for quantum computing consists of the one-qubit Hadamard gate H , the one-qubit T gate, and the two-qubit CNOT (controlled-NOT) gate [Chi22; Wol22]. These are specified by the following:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}, \quad \text{CNOT } |b_1\rangle |b_2\rangle = |b_1\rangle |b_2 \oplus b_1\rangle,$$

where $b_1, b_2 \in \{0, 1\}$. The resulting operations on N -qubit state $|\psi\rangle$ are then applying these gates to any qubit (or two qubits in the case of CNOT) of choice, and to act on the other qubits with the identity. The universality of the gate set then means that the subgroup of $U((\mathbb{C}^2)^{\otimes N})$ generated by these operations is dense.¹

Beyond the manipulation of the state of the quantum computer, one has to be able to provide *inputs* and *outputs*. We assume that the quantum computer is initialized in the state $|0\rangle$. The input for a computation can be encoded as part of a circuit (e.g., a sequence of NOT-gates at the start of the circuit), or using a special additional unitary operator, implementing a *quantum query* to the input. This usually comes in the form of a unitary operator that may be applied (as well as its inverse) with some prescribed behavior, without specifying how it is implemented in terms of standard gates. A common example is quantum query access to a bit string $x \in \{0, 1\}^N$ (or equivalently a Boolean function $f: [N] \rightarrow \{0, 1\}$): one is allowed to apply the unitary U_x on $\mathbb{C}^N \otimes \mathbb{C}^2$ defined by

$$U_x |i\rangle |b\rangle = |i\rangle |b \oplus x_i\rangle, \quad i \in [N], b \in \{0, 1\}.$$

The output, on the other hand, is quite different from the classical situation. At the end of the circuit, the state $|\psi\rangle$ is *measured*, and we assume for simplicity that this is a *standard basis measurement*. This means that the output will be a bit string $x \in \{0, 1\}^N$, such that x occurs with probability $|\langle x|\psi\rangle|^2$. Note that these probabilities add up to 1 because ψ has norm 1. This is a specific instance of *Born's rule*: if a quantum system is in a state $\rho = |\psi\rangle\langle\psi|$, and one has a collection of positive-semidefinite operators P_1, \dots, P_m such that $\sum_i P_i = I$, then the measurement corresponding to this shows outcome $i \in [m]$ with probability $\text{Tr}[P_i \rho]$.

It is by now well-known that certain tasks can be performed faster on a quantum computer than (known) on a classical computer. Two particularly famous examples are Shor's polynomial-time algorithm for factoring integers [Sho94], and Grover's algorithm for unstructured database search [Gro96]. The latter is particularly important for us, and informally the result is as follows: given quantum query access to a bit string $x \in \{0, 1\}^N$ with exactly one index $i \in [N]$ such that $x_i = 1$, one can find this i with probability $\geq 2/3$ using $O(\sqrt{N})$ queries to x , and $\tilde{O}(\sqrt{N})$ other gates.² Grover's algorithm for unstructured search has been generalized to that of *amplitude amplification* [BHMT02]; a related algorithm is *amplitude estimation*. We make use of these on a regular basis in the other chapters.

Although not used in the rest of this thesis, we mention several more modern algorithmic techniques. A particularly important one is the Harrow–Hassidim–Lloyd (HHL) algorithm [HHL09], which gives an exponential speedup for (sparse)

¹If one is only allowed to use the H -, CNOT- and $S = T^2$ -gates, then the resulting subgroup is finite and known as the Clifford group, see e.g. [NC02, Thm. 10.6].

²The number of gates here can be optimized further, see [AW17].

Subroutine AmpEst(U, M)

Input: Access to controlled versions of unitary $U \in U(2^q)$ and its inverse, an integer $M \geq 1$.

Output: Real number $\tilde{a} \in [0, 1]$.

Analysis: Theorem 11.2.2

linear system solving. This was improved through the introduction of variable-time amplitude amplification by Ambainis [Amb10]. Another is the use of *quantum walks* generalizing classical random walks, see e.g. [Amb07; Sze04]. These two admit a common generalization, namely that of the *quantum singular value transform* which has seen wide applications, see [GSLW19] and the references therein.

11.2. Common subroutines

In this section we summarize the external results that we build upon, and in some cases give a quick proof of an aspect of the result that is not mentioned explicitly in the original source.

11.2.1. Amplitude amplification and estimation

The following result can be derived from [BHMT02] (or [AR20] if one wants to avoid the quantum Fourier transform):

Theorem 11.2.1 (Amplitude amplification [BHMT02]). *Let $U \in \mathbb{C}^{2^q \times 2^q}$ be a unitary that creates a state*

$$|\psi\rangle = U|0^q\rangle = \sqrt{\alpha}|\phi_1\rangle|1\rangle + \sqrt{1-\alpha}|\phi_0\rangle|0\rangle$$

with $\alpha > 0$, and we know a lower bound $\alpha \geq \alpha' > 0$. Then there is a quantum algorithm V , implemented using $O(1/\sqrt{\alpha'})$ applications of U and U^\dagger , and $\tilde{O}(q/\sqrt{\alpha'})$ other elementary operations, such that

$$V|0^q\rangle = \sqrt{b}|\phi_0\rangle|0\rangle + \sqrt{1-b}|\phi_1\rangle|1\rangle, \quad \text{for some } b \in [1/2, 1].$$

Theorem 11.2.2 (Amplitude estimation [BHMT02, Thm. 12]). *Let $U \in \mathbb{C}^{2^q \times 2^q}$ be a unitary that creates a state*

$$|\psi\rangle = U|0^q\rangle = \sqrt{\alpha}|\phi_1\rangle|1\rangle + \sqrt{1-\alpha}|\phi_0\rangle|0\rangle.$$

There is a quantum algorithm **AmpEst** that, with probability $\geq \frac{8}{\pi^2}$, outputs an $\tilde{a} \in [0, 1]$ such that

$$|\alpha - \tilde{a}| \leq 2\pi \frac{\sqrt{\alpha(1-\alpha)}}{M} + \frac{\pi^2}{M^2}$$

using M applications of controlled- U and M applications of controlled- U^\dagger . If M is a power of 2, the algorithm uses $O(qM)$ additional quantum gates, and the computation of the sine-squared function of the normalized phase.

Subroutine GroverCertainty(U, k_0)

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$, an integer $k_0 \geq 1$.**Output:** An index $i \in [N]$.**Guarantee:** If $|x| = k_0$, then $x_i = 1$ with certainty.**Analysis:** Theorem 11.2.3

Proof. This follows from the formulation in [BHMT02] by setting $k = 1$ and implementing the reflection through $|0^q\rangle$ using $O(q)$ gates, which needs to be performed M times. If M is a power of 2 we can implement the quantum Fourier transform on $m = \log_2(M)$ qubits using m Hadamard gates, and the QFT and its inverse need only be performed once; therefore, this cost is absorbed in the big- O . \square

We note that the above formulation of `AmpEst` outputs a real number \tilde{a} whereas we require a fixed-point encoded number for future uses. However, it suffices to use fixed-point arithmetic using $O(\log(M))$ bits; after all, the guarantee of `AmpEst` only gives a precision of $1/\text{poly}(M)$.

11.2.2. Variations of Grover search

We also need a version of amplitude amplification (Theorem 11.2.1) where the success probability is 1 if one knows the amplitude of the “good” part of the state exactly. In a nutshell, the algorithm with success probability 1 is the usual amplitude amplification algorithm applied not to U but to U followed by a rotation of the last qubit to slightly reduce the amplitude a to \tilde{a} . Carefully choosing \tilde{a} ensures that the success probability is exactly 1 after an integer number of rounds of amplitude amplification. This requires having access to gates which implement rotation by arbitrary angles, not just angles of the form $\pi/2^m$ for some integer m . We specialize the statement of this result to the search setting but remark that this works more generally for amplitude amplification. For exactly $N/4$ marked elements this observation was first made in [BBHT98].

Theorem 11.2.3 ([BHMT02, Thm. 4]). *Let $x \in \{0, 1\}^N$ with $|x| = k \geq 1$. Then there is a quantum algorithm `GroverCertainty` that takes as input a quantum oracle U_x to access x and an integer $k_0 \in [N]$, and that outputs an index $i \in [N]$, such that $x_i = 1$ with certainty if $k_0 = k$, and uses $O(\sqrt{N/k_0})$ quantum queries to x , and $O(\sqrt{N/k_0} \log(N))$ additional gates.*

One can use the above to find all k indices $i \in [N]$ such that $x_i = 1$, with probability 1, using $O(\sqrt{Nk})$ queries and $\tilde{O}(\sqrt{Nk}^{3/2})$ gates. We give a precise statement and implementation of this in Lemma 12.3.2.

The other version of Grover that we need is the following, which is originally due to [BBHT98, Thm. 3], but we use a slightly different version from [BHMT02, Thm. 3]:

Theorem 11.2.4. *Let $x \in \{0, 1\}^N$ with $|x| = k$, where k is not necessarily known. Then there is a quantum algorithm `GroverExpectation` that takes as input a quantum oracle U_x to access x , and if $k \geq 1$, outputs an index $i \in [N]$ such that $x_i = 1$. The number of quantum queries to x that it uses is a random variable Q , such that, if $k \geq 1$, then*

Subroutine GroverExpectation(U_x)

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$.**Output:** An index $i \in [N]$.**Guarantee:** If $|x| \geq 1$, then $x_i = 1$ with certainty.**Analysis:** Theorem 11.2.4

Subroutine Grover_{2/3}(U_x, k_{lb})

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$ and a lower bound k_{lb} on $|x|$.**Output:** An index $i \in [N]$.**Guarantee:** If $|x| \geq 1$, then with probability $\geq 2/3$, $x_i = 1$.**Analysis:** Lemma 11.2.5

$$\mathbb{E}[Q] = O\left(\sqrt{N/k}\right),$$

and if $|x| = 0$, then $Q = \infty$ (i.e., the algorithm runs forever). The number of additional gates used is $O(Q \log(N))$. The index i which is output is uniformly random among all such indices, and independent of the value of Q .

Lemma 11.2.5. Let $x \in \{0, 1\}^N$. Then there is a quantum algorithm Grover_{2/3} that takes as input a quantum oracle U_x to access x and a lower bound $k_{lb} \geq 1$ on $|x|$. With probability $\geq 2/3$, it outputs an index $i \in [N]$ such that $x_i = 1$. It uses $O(\sqrt{N/k_{lb}})$ quantum queries to x , and $O(\sqrt{N/k_{lb}} \log(N))$ additional gates.

Proof. The algorithm GroverExpectation finds an index i such that $x_i = 1$. Its number of applications of controlled- U_x is a random variable Q and the number of additional gates is $O(Q \cdot \log(N))$. By Theorem 11.2.4 we have $\mathbb{E}[Q] = O(\sqrt{N/|x|})$. Markov's inequality shows that if we terminate GroverExpectation after at most $C\sqrt{N/|x|}$ quantum queries for a suitable constant $C > 0$, then it finds an index i such that $x_i = 1$ with probability at least $2/3$. The procedure Grover_{2/3} uses the lower bound k_{lb} on $|x|$ to decide when to terminate GroverExpectation. For the same constant $C > 0$ as before, it terminates after at most $C\sqrt{N/k_{lb}}$ quantum queries. Since $C\sqrt{N/k_{lb}} \geq C\sqrt{N/|x|}$, the success probability of Grover_{2/3} is also at least $2/3$. \square

Let us make some remarks about the complexity of finding a single marked element. First, to find such an element with certainty one can essentially remove the $\log(N)$ factor in the gate complexity: $O(\sqrt{N} \log(\log^*(N)))$ gates suffice [AW17]. Second, by cleverly combining GroverCertainty and Grover_{2/3}, one can find a marked element (among an unknown number of solutions) with probability $\geq 1 - \rho$ using $\sqrt{N} \log(1/\rho)$ quantum queries [BCWZ99]. This shows that the standard way of boosting the success probability of Grover_{2/3} is not optimal.

Using GroverExpectation as a subroutine, one can find the index of a minimum (or maximum) of a function $[N] \rightarrow \mathbb{R}$ in roughly \sqrt{N} time. More generally, for a totally ordered finite set S , one can find a maximal element:

Subroutine $\text{ApproxCount}(U_x, \varepsilon, \rho)$

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$, rational number $\varepsilon > 0$ such that $\frac{1}{3N} < \varepsilon \leq 1$, failure probability $\rho > 0$.

Output: Integer $\tilde{k} \in \{0, \dots, N\}$.

Guarantee: If $|x| = k \geq 1$, with probability $\geq 1 - \rho$, $|\tilde{k} - k| \leq \varepsilon k$, and if $k = 0$ then $\tilde{k} = 0$ with certainty.

Analysis: Theorem 11.2.7

Theorem 11.2.6 (Quantum max/min-finding [DHHM06]). Let $S = \{s_1, \dots, s_N\}$ be a finite set of size $|S| = N$, endowed with a total order \leq . Suppose we have a unitary U_{\leq} acting on $\mathbb{C}^N \otimes \mathbb{C}^N \otimes \mathbb{C}^2$ such that

$$U_{\leq} |i\rangle |j\rangle |b\rangle = |i\rangle |j\rangle |b \oplus (s_i \leq s_j)\rangle.$$

Then there exists a quantum algorithm that finds, with probability $\geq 2/3$, an index $i \in [N]$ such that $s \leq s_i$ for all $s \in S$, using $O(\sqrt{N})$ queries to U_{\leq} and $\tilde{O}(\sqrt{N})$ other gates.

11.2.3. Approximate counting and summation

Next, we recall a well-known result on approximate counting.

Theorem 11.2.7 ([BHMT02, Thm. 18]). Let $x \in \{0, 1\}^N$ and write $|x| = k$. Let $\frac{1}{3N} < \varepsilon \leq 1$. Then there is a quantum algorithm that, with probability at least $2/3$, that outputs an estimate \tilde{k} such that

$$|\tilde{k} - k| \leq \varepsilon k$$

using an expected number of

$$\Theta \left(\sqrt{\frac{N}{\lfloor \varepsilon k \rfloor + 1}} + \frac{\sqrt{k(N-k)}}{\lfloor \varepsilon k \rfloor + 1} \right)$$

quantum queries to x . If $k = 0$, then the algorithm outputs $\tilde{k} = 0$ with certainty, using $\Theta(\sqrt{N})$ quantum queries to x . In both cases, the algorithm uses a number of gates which is $O(\log(N))$ times the number of quantum queries. To boost the success probability to $1 - \rho$, repeat the procedure $O(\log(1/\rho))$ many times and output the median of the returned values.

We often use the special case $\varepsilon = 1/2$ of the above theorem, hence we record it here for future use. (Note that the proof of Theorem 11.2.7 given in [BHMT02] in fact starts by obtaining a constant factor approximation of $|x|$.)

Corollary 11.2.8. Let $x \in \{0, 1\}^N$ and write $|x| = k$. Then there is a quantum algorithm that outputs a k_{est} such that, with probability $\geq 1 - \rho$, we have $k/2 \leq k_{\text{est}} \leq 3k/2$, and uses $O(\sqrt{N}/(k+1) \log(1/\rho))$ quantum queries and $O(\sqrt{N}/(k+1) \log(1/\rho) \log(N))$ gates.

We now discuss known extensions of the above results on counting the Hamming weight of a bit string to the problem of *mean estimation*: given a vector $v \in [0, 1]^N$, one is interested in approximating $\bar{v} = \frac{1}{N} \sum_{i=1}^N v_i$. This was first studied in [Gro97] and later in [Gro98] where in the latter it was shown that one can find an additive

ε -approximation of \bar{v} using $\tilde{O}(1/\varepsilon)$ quantum queries to a unitary that prepares a state encoding the entries of v in its amplitudes, and a similar number of additional gates (also dependent on N). Using amplitude amplification techniques one can reduce the query dependence to $O(1/\varepsilon)$ with $O(\log(N)/\varepsilon)$ additional gates. This result may be easily recovered from Theorem 11.2.2 with $M = \Theta(1/\varepsilon)$, applied to a unitary preparing

$$\frac{1}{\sqrt{N}} \sum_{i=1}^N |i\rangle (\sqrt{1-v_i} |0\rangle + \sqrt{v_i} |1\rangle).$$

It is well-known that when one has quantum oracle access to fixed point representations of the entries of v (cf. Definition 12.2.2), rather than just a state encoding its entries in the amplitudes, one can give an algorithm whose complexity depends only on N and δ , with guarantees as given below.

Theorem 11.2.9. *Let $v \in [0, 1]^N$ be a vector with each entry v_i encoded in $(0, b)$ -fixed-point format, and let U_v be a unitary implementing binary oracle access to v (cf. Definition 12.2.2).*

Let $\rho \in (0, 1)$. Then with $O(\frac{\sqrt{N}}{\delta} \log(1/\rho))$ applications of controlled- U_v , controlled- U_v^\dagger , and a polylogarithmic gate overhead, one can find with probability $\geq 1 - \rho$ a multiplicative δ -approximation of $\frac{1}{N} \sum_i v_i$.

We shall give an improvement over this in Chapter 12.

11.3. Lower bound techniques

An important property of the theory of quantum *query* algorithms is that it is possible to prove interesting *query lower bounds*, precisely because sometimes one can obtain algorithms with a query complexity smaller than the natural input size. For instance, in case of Grover's algorithm for unstructured search on a bit string of size N , one can prove that its query complexity of $O(\sqrt{N})$ is asymptotically optimal [BBBV97]. Although various tools are now available for proving quantum query lower bounds (see e.g. [BBBV97; BBC+01; AMRR11; Bel23]), we shall only use one in the rest of this thesis: a version of Ambainis' adversary bound [Amb02]. We record the (complicated) statement here:

Lemma 11.3.1 ([Amb02, Thm. 6.1]). *Let $f: A \subseteq \Sigma^N \rightarrow B$ be a function of N variables, which takes values in some finite set B . Let $X, Y \subseteq A$ be two sets of inputs such that $f(x) \neq f(y)$ if $x \in X$ and $y \in Y$. Let $R \subseteq X \times Y$ be nonempty, and suppose that it satisfies:*

- *For every $x \in X$, there exist at least m_X different $y \in Y$ such that $(x, y) \in R$.*
- *For every $y \in Y$, there exist at least m_Y different $x \in X$ such that $(x, y) \in R$.*

Let $\ell_{x,i}$ be the number of $y \in Y$ such that $(x, y) \in R$ and $x_i \neq y_i$, and similarly for $\ell_{y,i}$. Let $\ell_{\max} = \max_{i \in [N]} \max_{(x,y) \in R, x_i \neq y_i} \ell_{x,i} \ell_{y,i}$. Then any algorithm that computes f with success probability $\geq 2/3$ uses $\Omega\left(\sqrt{\frac{m_X m_Y}{\ell_{\max}}}\right)$ quantum queries to the input.

Intuitively, the goal is to choose relation R in the theorem is chosen such that the related inputs in X and Y are *hard* to distinguish using a single query. At least $m_X m_Y$ pairs of (x, y) need to be distinguished from one another. For a fixed

11. *An introduction to quantum algorithms and lower bounds*

input $x \in X$, $\ell_{x,i}$ is the number of related inputs in Y that can be distinguished by querying the i -th component of x . Therefore ℓ_{\max} gives a quantitative interpretation of the “maximum distinguishing power” of a single query.

We remark here that the adversary method can be generalized to (negatively) weighted versions (whereas the above has a combinatorial flavor), and that such generalizations in fact even *characterize* the quantum query complexity of a function. We refer to [Wol22] for more information.

12. Basic quantum subroutines, improved

In this chapter we consider two problems. The first problem is to find all marked indices of a given bit string. We give a quantum algorithm that has optimal query complexity and only polylogarithmic overhead in the gate complexity, in the setting where one only has a small quantum memory; previous approaches either had logarithmic overhead in the query complexity or a polynomial overhead in the gate complexity.

The second problem is to compute a multiplicative approximation of a sum of non-negative numbers, given quantum query access to binary descriptions of these numbers. For this problem, we give a quantum algorithm that has a quadratically better dependence on the approximation error than the best previously known approach.

We first provide a more detailed introduction in Section 12.1. In Section 12.2 we set our notation and define the input model. In Section 12.3 we provide our algorithm for searching for multiple marked elements. Lastly, in Section 12.4, we give our summation algorithm.

12.1. Introduction

12.1.1. Finding multiple marked elements in a list

Grover's famous search algorithm [Gro96] can be used to find a marked element in a list quadratically faster than possible classically. Formally it can be used to solve the following problem: given a bit string $x \in \{0, 1\}^N$, $x \neq 0$, find an index $i \in [N]$ such that $x_i = 1$.

In this work we consider the problem of finding *all* indices $i \in [N]$ for which $x_i = 1$. We give a query-optimal quantum algorithm with polylogarithmic gate overhead in the setting where one has a *small quantum memory*. We explain below why this last assumption makes the problem non-trivial. This improves over the previous state-of-the-art: previous algorithms were either query-optimal but with a polynomial gate overhead, or had a polylogarithmic gate overhead but also a logarithmic overhead in the query count.

A well-known query-optimal algorithm for the problem is as follows [GW02, Lem. 2]. Let k be the Hamming weight $|x| := \sum_{i=1}^N x_i$ of x . For ease of exposition, suppose the algorithm knows k . (For our results we will work with weaker assumptions such as knowing only an upper bound on k , or an estimate of it, see Section 12.3. We also ignore failure probabilities in this part of the introduction.) A variant of Grover's algorithm [BBHT98] can find a single marked element

This chapter is adapted from [AGN23].

12. Basic quantum subroutines, improved

using $O(\sqrt{N/k})$ quantum queries and $O(\sqrt{N/k} \log(N))$ additional single- and two-qubit gates. One can then find all k marked elements using

$$O\left(\sqrt{N/k} + \sqrt{N/(k-1)} + \dots + \sqrt{N}\right) = O\left(\sqrt{Nk}\right)$$

quantum queries to x . The above complexity is obtained as follows. Suppose we have already found a set $J \subseteq [N]$ of marked elements. Then to find a new marked element, we replace x by the string $z \in \{0, 1\}^N$ defined as

$$z_i = \begin{cases} x_i & \text{if } i \notin J, \\ 0 & \text{otherwise.} \end{cases}$$

A quantum query to z can be made using a single quantum query to x and quantum query to J (which on input $|i\rangle |b\rangle$ for $i \in [N], b \in \{0, 1\}$ returns $|i\rangle |b \oplus \delta_{i \in J}\rangle$ where $\delta_{i \in J} \in \{0, 1\}$ is one iff $i \in J$). In particular, if J can be stored in a quantum memory (i.e. queried and updated in unit time), then the query complexity will be $O(\sqrt{Nk})$ and the time complexity is $\tilde{O}(\sqrt{Nk})$. We refer the interested reader to [GLM08] and [CHI+18, Sec. 5] for a discussion of quantum memory and its (dis)advantages.

However, when we cannot store J in a quantum memory, a naive implementation of the quantum queries to J is expensive in terms of gate complexity: if $|J| = s$, then one can use $O(s \log(N))$ quantum gates to implement a single query to J .¹ Since the size of J grows to k , the total gate complexity of finding all marked elements will scale as $\tilde{O}(\sqrt{Nk}^{3/2})$, which is a factor k larger than the query complexity. We show that this factor of k in the gate complexity can be avoided: we give an algorithm that finds, with large probability, all k indices using the optimal number of quantum queries to x , $O(\sqrt{Nk})$, while incurring only a polylogarithmic overhead in the gate complexity, in the case where we only have a small quantum memory. We state a simplified version of our main result below; for the full version, see Theorem 12.3.9 and the corresponding algorithm `GroverMultipleFast`.

Theorem 12.1.1. *Let $x \in \{0, 1\}^N$ with $|x| = k \geq 2$, and let $\rho \in (0, 1)$ be such that $\rho = \Omega(1/\text{poly}(k))$. Then we can find, with probability $\geq 1 - \rho$, all k indices $i \in [N]$ for which $x_i = 1$ using $O(\sqrt{Nk})$ quantum queries and $O(\sqrt{Nk} \log(k)^3 \log(N))$ additional gates.*

We mention that by a simple coupon-collector argument one can already achieve both query- and gate-complexity $\sqrt{Nk} \text{polylog}(N, 1/\rho)$, see Proposition 12.3.7. Our algorithm completely removes the $\text{polylog}(N)$ factors in the query complexity and moreover has a much improved dependence on $\log(1/\rho)$: one can achieve $\rho = 1/\text{poly}(k)$ without increasing the number of quantum queries made by the algorithm. In the same spirit, we mention that previous work had already shown that simply boosting a constant success probability is not optimal for finding a single marked element: one can do so with probability $\geq 1 - \rho$ using $\sqrt{N \log(1/\rho)}$ quantum queries [BCWZ99].

In a nutshell, our algorithm is a hybrid between the quantum coupon-collector and the query-optimal algorithm described above. First, we use the coupon

¹We ignore here the cost of maintaining a classical data structure for J , but comment on this again later.

collection strategy to find t marked indices $1 \leq i_1 < \dots < i_t \leq n$, for t roughly $k/\log(k)^2$. A basic property of this strategy is that the resulting indices $\{i_1, \dots, i_t\}$ yield a uniformly random subset of size t of the marked indices in x . Next, for every $j \in [t + 1]$, we use the query-optimal algorithm to find all remaining marked elements in the interval $(i_{j-1}, i_j) \subseteq [n]$, where we write $i_0 = 0$ and $i_{t+1} = n + 1$. With high probability over the found indices $\{i_1, \dots, i_t\}$, each of the intervals (i_{j-1}, i_j) contains few remaining marked indices, which reduces the effect of the high gate-complexity overhead of the query-optimal search algorithm.

12.1.2. Improved quantum summing algorithm

Given quantum query access to a binary description of $v \in [0, 1]^N$, how difficult is it to obtain, with probability $\geq 1 - \rho$, a $(1 \pm \delta)$ -multiplicative approximation² of the sum $s = \sum_{i=1}^N v_i$? We provide an algorithm to do so whose complexity can be tuned by choosing a parameter $p \in (0, 1)$; one special case of our second main result is as follows, see Theorem 12.4.3 for the full version. In the version below we have made very mild assumptions on the failure probability ρ and precision δ , which essentially correspond to the regime in which one makes at most $O(N)$ quantum queries.

Theorem 12.1.2 (Informal version of Theorem 12.4.3). *Let $v \in [0, 1]^N$. Let $\rho, \delta \in (0, 1)$ be such that $\log(1/\rho)/\delta = O(N)$. Then we can find, with probability $\geq 1 - \rho$, a $(1 \pm \delta)$ -multiplicative approximation of $\sum_{i=1}^N v_i$ using*

$$O\left(\sqrt{\frac{N}{\delta} \log(1/\rho)}\right) \quad (12.1.1)$$

quantum queries to binary descriptions of the entries of v , and a gate complexity which is larger by a factor polylogarithmic in N , $1/\delta$ and $1/\rho$.

In a nutshell, our algorithm first finds all indices of “large enough” entries of the v using `GroverMultipleFast` and sums the corresponding elements classically. It then rescales the remaining “small enough” elements and uses amplitude estimation [BHMT02] to approximate their sum. To determine what “large enough” means, we use a recent quantum quantile estimation procedure from [Ham21]. Choosing the quantile carefully controls both the number of elements that need to be found in the first stage, as well as the size of the elements that remain to be summed in the second stage. Note that it is the above version of Grover’s algorithm that allows us to obtain a query complexity with only a $\sqrt{\log(1/\rho)}$ -dependence, and without additional polylogarithmic factors in N and δ . Indeed, the fact that the number of quantum queries required to find multiple marked elements does not depend on $\log(1/\rho)$ (for ρ not too small) allows us to balance the complexities of the two stages.

The problem we consider can be viewed as a special case of the mean estimation problem, or as a generalization of the approximate counting problem for binary strings $x \in \{0, 1\}^N$. We briefly discuss how our results compare to prior work on those problems.

²Here we use the convention that a $(1 \pm \delta)$ -multiplicative approximation of a real number $s \geq 0$ is a real number $\tilde{s} \geq 0$ for which $(1 - \delta)s \leq \tilde{s} \leq (1 + \delta)s$.

Mean estimation algorithms. After multiplying the v_i by a factor $\frac{1}{N}$, we can interpret the problem of finding a $(1 \pm \delta)$ -multiplicative approximation of the sum $s = \sum_{i=1}^N v_i$ as the problem of obtaining a $(1 \pm \delta)$ -multiplicative approximation of the mean $\mu = \frac{1}{N} \sum_{i=1}^N v_i$ of the random variable that, for each $i \in [N]$, takes value v_i with probability $1/N$. Quantum algorithms for the mean estimation problem date back to the work of Grover [Gro97; Gro98]. A careful application of maximum finding and quantum amplitude estimation yields such an approximation of μ , with probability $\geq 1 - \rho$, using $O(\frac{\sqrt{N}}{\delta} \log(1/\rho))$ quantum queries and polylogarithmic gate overhead, see Theorem 11.2.9. We improve the dependence on δ from $1/\delta$ to $1/\sqrt{\delta}$.

We remark that faster mean estimation algorithms have been developed for example for random variables with a small variance σ^2 . Indeed, the current state of the art obtains a $(1 \pm \delta)$ -multiplicative approximation, with probability $\geq 1 - \rho$, using $\tilde{O}((\frac{\sigma}{\delta\mu} + \frac{1}{\sqrt{\delta\mu}}) \log(1/\rho))$ quantum queries in expectation [Ham21; KO23].³

For comparison, we mention that $\sigma \leq \sqrt{\mu(1-\mu)}$ always holds, and when given binary access to the v_i , one may additionally assume (after maximum-finding and rescaling) that $\mu \in [1/n, 1]$.

Approximate counting algorithms. As mentioned above, our algorithm improves the error-dependence for mean estimation (for random variables with small support). It therefore makes sense to compare our upper bound with the well-known lower bound for the approximate counting problem for binary strings $x \in \{0, 1\}^N$. We first recall a precise statement. Let $x \in \{0, 1\}^N$ and $k = |x|$, and U_x a unitary implementing quantum oracle access to x . Then for an integer $\Delta > 0$, any quantum algorithm which, with probability $\geq 2/3$, computes an additive Δ -approximation of k uses at least $\Omega(\sqrt{N/\Delta} + \sqrt{k(N-k)}/\Delta)$ applications of controlled- U_x [NW99, Thm. 1.10 and 1.11]. A matching upper bound is given in [BHMT02, Thm. 18], see Theorem 11.2.7 for a precise formulation. We can compare the complexity of our algorithm by converting multiplicative error into additive error, i.e., to achieve an additive error of ε we take $\delta = \varepsilon/k$ (or ε divided by a suitable multiplicative approximation of k). Then the key point is that if one considers Eq. (12.1.1) for $\varepsilon \leq \Delta$ and $k \geq 1$, then

$$\sqrt{\frac{Nk}{\varepsilon}} \geq \sqrt{\frac{Nk}{\Delta}} \geq \sqrt{\frac{1}{2} \frac{N}{\Delta} + \frac{1}{2} \frac{k(N-k)}{\Delta}} \geq \frac{1}{2} \left(\sqrt{\frac{N}{\Delta}} + \frac{\sqrt{k(N-k)}}{\Delta} \right)$$

where the last inequality follows from concavity of the square-root function and $\Delta \geq 1$. In other words, for all parameters N, k, Δ , the complexity of our algorithm is at least as large as the lower bound on approximate counting. In particular, when Δ is large, our bound is suboptimal for quantum counting. This is no surprise given that our algorithm finds all “large” elements, which in the counting setting would amount to finding all ones. On the other hand, when Δ is a small constant, the approximate counting lower bound shows that our upper

³In [Ham21, Proposition 6.4], a matching (up to log-factors) lower bound is shown for Bernoulli random variables. We remark that our algorithm does not break that lower bound since we parameterize the problem differently: the complexity of our algorithm depends also on the size of the support of the distribution.

bound is essentially tight. We leave it as an open problem whether one can obtain a quantum algorithm for approximate summing of vectors $v \in [0, 1]^N$ that matches the approximate counting complexity when applied to $v \in \{0, 1\}^N$ for the entire range of parameters N, k, Δ .

12.2. Preliminaries

12.2.1. Notation and assumptions

Throughout this section, we will assume that $N = 2^n \geq 1$ for some $n \geq 1$. We identify \mathbb{C}^N with \mathbb{C}^{2^n} by $|j\rangle \mapsto |j_1 \dots j_n\rangle$, where $(j_1, \dots, j_n) \in \{0, 1\}^n$ is the standard binary encoding of $j - 1 \in \{0, \dots, 2^n - 1\}$. We write \log_2 for the logarithm with base 2 and \ln for the natural logarithm. For a bit string $x \in \{0, 1\}^N$ we write $|x| = \sum_{i \in [N]} x_i$. Throughout we will use k to denote the Hamming weight of x , i.e., $|x| = k$, and we write $k_{\text{est}}, k_{\text{lb}}, k_{\text{ub}}$ for various bounds on k : k_{est} will denote an integer such that $k/2 \leq k_{\text{est}} \leq 3k/2$, and k_{lb} and k_{ub} are lower- and upper bounds on k respectively.

12.2.2. Computational model

We express the cost of a quantum algorithm in terms of the number of one- and two-qubit gates it uses. Note that in particular we allow single-qubit rotations with arbitrary real angles. In Section 12.3, the angle will always be determined by classical data. On the other hand, in Section 12.4, we use only angles of the form $\pi/2^m$ and carefully count the number of used gates. The reason for the slightly different approach in the latter situation is that the total angle to be used will be in superposition (i.e., depend on a binary description in another register). In the query setting, we separately count the number of quantum queries the algorithm makes, which means (controlled) applications of the query unitary or its inverse. We will use the following types of quantum queries to access either N -bit strings $x \in \{0, 1\}^N$ or N -dimensional vectors $v \in [0, 1]^N$ (specified in fixed-point format).

Definition 12.2.1. A unitary $U_x \in U(\mathbb{C}^N \otimes \mathbb{C}^2)$ is said to implement quantum oracle access to an N -bit string $x \in \{0, 1\}^N$ if it acts as

$$U_x |i\rangle |b\rangle = |i\rangle |b \oplus x_i\rangle$$

for all $i \in [N]$ and $b \in \{0, 1\}$.

Definition 12.2.2. A unitary $U_v \in U(\mathbb{C}^N \otimes \mathbb{C}^{2^b})$ is said to implement quantum oracle access to $(0, b)$ -fixed-point representations of $v \in [0, 1]^N$ if it acts as

$$U_v |i\rangle |0^b\rangle = |i\rangle |v_i\rangle$$

for all $i \in [N]$, where $|v_i\rangle = |(v_i)_1 \dots (v_i)_b\rangle$ satisfies $\sum_{j=1}^b (v_i)_j 2^{-j} = v_i$.

In both cases we allow the unitary to act on additional workspace registers, which we omit for notational convenience.

We additionally use a classical data structure to maintain sorted lists that supports both insertion and look-up in a time that scales logarithmically with the size of the list, see for example [Knu98, Sec. 6.2.3] or [CLRS22, Ch. 13].

12.3. Fast Grover search for multiple items, without quantum memory

In this section we give a version of Grover’s search algorithm for the problem of, given a string $x \in \{0,1\}^N$, finding *all* k indices $i \in [N]$ such that $x_i = 1$. For $\rho \in (0,1)$ with $\rho = \Omega(1/\text{poly}(k))$, our algorithm finds all such indices with probability $\geq 1 - \rho$, and uses $O(\sqrt{Nk})$ quantum queries and $\tilde{O}(\sqrt{Nk})$ single- and two-qubit gates. The contribution here is that the query complexity is optimal and the time complexity is only polylogarithmically worse than the query complexity, without using a QRAM.

12.3.1. Deterministic Grover for multiple elements

We first recall the well-known result [GW02, Lem. 2], that it is possible to find all solutions with probability 1 using $O(\sqrt{Nk})$ quantum queries, which is optimal, but suffers from a too-high gate complexity in terms of k . The algorithm is given in `GroverCertaintyMultiple`. We first define for each $j \in [N]$ a gate C_j , referred to as the “control-on- j -NOT”-gate, and describe how to implement it with a standard gate-set. The point of this gate is that if one has quantum oracle access U_x to $x \in \{0,1\}^N$, then $C_j U_x$ implements quantum oracle access to the bit string $y \in \{0,1\}^N$ which agrees with x on all indices, except on the j -th index, where the bit is flipped.

Lemma 12.3.1. *Let $N = 2^n$. For $j \in [N]$ define the “control-on- j -NOT”-gate $C_j \in U(\mathbb{C}^N \otimes \mathbb{C}^2)$ by*

$$C_j |i\rangle |b\rangle = \begin{cases} |i\rangle |b \oplus 1\rangle & \text{if } i = j, \\ |i\rangle |b\rangle & \text{otherwise.} \end{cases} \quad (12.3.1)$$

Then the C_j -gate can be implemented with $O(n)$ standard gates and $n - 1$ ancillary qubits.

Proof. Let $|j\rangle = |j_1 \dots j_n\rangle$ be the binary encoding of $j - 1$. Then:

- (i) For each $l \in [n]$ such that $j_l = 0$, apply a NOT gate on the l -th qubit of the index register.
- (ii) Apply a NOT-gate to the output register containing b , controlled on all n qubits of the index register. This can be implemented using $O(n)$ Toffoli gates, one CNOT gate, and $n - 1$ ancilla qubits, see [NC02, Fig. 4.10].
- (iii) Apply the NOT gates from the first step again. □

Lemma 12.3.2. *Let $x \in \{0,1\}^N$, U_x a quantum oracle to access x , and $k_{\text{ub}} \geq 1$. If $|x| = k \leq k_{\text{ub}}$, then `GroverCertaintyMultiple`(U_x, k_{ub}) finds, with probability 1, all k indices i such that $x_i = 1$. The algorithm uses*

$$O\left(\sqrt{Nk_{\text{ub}}}\right)$$

applications of U_x , and

$$O\left(\sqrt{Nk_{\text{ub}}}(k + 1) \log(N)\right)$$

additional non-query gates.

Subroutine GroverCertaintyMultiple(U_x, k_{ub})

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$, an integer $k_{ub} \geq 1$.**Output:** Classical list of indices $J \subseteq [N]$.**Guarantee:** If $|x| \leq k_{ub}$, then for every $j \in [N]$, $j \in J$ if and only if $x_j = 1$.**Analysis:** Lemma 12.3.2

```

1  $J_{k_{ub}} \leftarrow \emptyset;$ 
2  $U_{J_{k_{ub}}} \leftarrow U_x;$ 
3  $m \leftarrow k_{ub};$ 
4 while  $m > 0$  do
5   use GroverCertainty( $U_{J_m}, m$ ) to find a  $j \in [N] \setminus J_m;$ 
6   if  $x_j = 1$  then
7      $J_{m-1} \leftarrow J_m \cup \{j\};$ 
8      $U_{J_{m-1}} \leftarrow C_j U_{J_m}$ , where  $C_j$  is defined in Lemma 12.3.1;
9   else
10     $J_{m-1} \leftarrow J_m;$ 
11     $U_{J_{m-1}} \leftarrow U_{J_m};$ 
12  end if
13   $m \leftarrow m - 1;$ 
14 end while
15 return  $J_0;$ 

```

Proof. We first establish correctness of GroverCertaintyMultiple. For $m \in [k_{ub}]$, let $J_m \subseteq [N]$ be the index set and U_{J_m} the unitary in the algorithm at the m -th step. Then by the definition of C_j , U_{J_m} implements oracle access to the bit string y^m which agrees with x on $[N] \setminus J_m$, and is zero on the indices in J_m (whereas $x_j = 1$ for $j \in J_m$). Clearly $j \in J_0$ implies that $x_j = 1$. It remains to show that in k_{ub} iterations we find *all* marked elements. To do so, observe that there can be at most $k_{ub} - k$ iterations in which one fails to find a new $j \in [N]$ such that $x_j = 1$: indeed, as soon as this happens, we have $m = |y^m|$, and every iteration afterwards we find a new index with certainty by the guarantees of GroverCertainty.

In total, this procedure uses $\sum_{m=1}^{k_{ub}} O(\sqrt{N/m}) = O(\sqrt{Nk_{ub}})$ applications of U_x . The number of auxiliary gates for a single query in the m -th iteration is $O(|J_m| \cdot \log(N))$, and GroverCertainty itself uses an additional $O(\sqrt{N/m} \log(N))$ additional gates. Therefore the total number of gates in the m -th iteration is

$$O\left(\sqrt{N/m} \cdot |J_m| \cdot \log(N) + \sqrt{N/m} \log(N)\right) = O\left(\sqrt{N/m}(k+1) \log(N)\right)$$

Summing this over all iterations yields a total gate complexity of $O(\sqrt{Nk_{ub}}(k+1) \log(N))$. \square

12.3.2. Coupon collecting Grover

We next give another simple version of Grover which can be used to find a large fraction of the marked elements in a time-efficient manner, but does not yield a query-optimal bound when the fraction is close to 1. The algorithm is given in GroverCoupon, and is analyzed in Proposition 12.3.7.

12. Basic quantum subroutines, improved

The algorithm is simple: the idea is to repeatedly call $\text{Grover}_{2/3}$ to sample marked elements. The analysis is based on the observation that the required number of calls to $\text{Grover}_{2/3}$ is a sum of geometrically distributed random variables: for $1 \leq i \leq t$, the number of calls to obtain the i -th distinct marked element is a geometrically distributed random variable with success probability $p'_i \geq \frac{2}{3}p_i$, where $p_i = (k - i + 1)/k$ is the probability of observing a new element after $i - 1$ distinct elements have been found. This is because $\text{Grover}_{2/3}$ succeeds with probability $\geq 2/3$, and the fact that if $\text{Grover}_{2/3}$ successfully finds a marked element, then it is uniformly random among the marked elements. The number of calls can then be bounded using a general tail bound on sums of geometrically distributed random variables given in [Jan18, Thm. 2.3] (see Lemma 12.3.4).

The analysis is based on tail bounds of sums of geometrically distributed random variables. These tail bounds in turn are stated in terms of the harmonic numbers, for which we recall some basic properties in the following lemma.

Lemma 12.3.3. *The k -th harmonic number H_k is defined by $H_k = \sum_{j=1}^k \frac{1}{j}$, and we shall use the convention $H_0 = 0$. For $k \geq 1$ it satisfies*

$$H_k - \gamma - \ln(k) \in \left[\frac{1}{2(k+1)}, \frac{1}{2k} \right],$$

where $\gamma \approx 0.577$ is the Euler–Mascheroni constant. Furthermore, for $0 \leq t < k$, this implies

$$H_k - H_{k-t} \leq \ln\left(\frac{k}{k-t}\right) + \frac{2k-t+1}{2k(k-t+1)},$$

which in turn for $t \leq k/2 < k$ implies

$$H_k - H_{k-t} \leq \frac{2(t+1)}{k}.$$

Proof. The bounds on $H_k - \gamma - \ln(k)$ are well-known, see [You91] for an elementary proof. For the estimate on $H_k - H_{k-t}$ we have

$$H_k - H_{k-t} \leq \ln(k) - \ln(k-t) + \frac{1}{2k} + \frac{1}{2(k-t+1)} = \ln\left(\frac{k}{k-t}\right) + \frac{2k-t+1}{2k(k-t+1)}.$$

Furthermore, if $t \leq k/2 < k$, then

$$\ln\left(1 + \frac{t}{k-t}\right) + \frac{2k-t+1}{2k(k-t+1)} \leq \frac{2t}{k} + \frac{2k+1}{2k(k/2+1)} \leq \frac{2t}{k} + \frac{2}{k} = \frac{2(t+1)}{k}. \quad \square$$

We use the following tail bound for geometrically distributed variables.

Lemma 12.3.4 ([Jan18, Thm. 2.3]). *For $i \in [n]$ assume $X_i \sim \text{Geo}(p_i)$ for $p_i \in (0, 1]$. Let $X = \sum_{i \in [n]} X_i$ and write $\mu = \mathbb{E}[X]$, $p_* = \min_{i \in [n]} p_i$. Then for any $\lambda \geq 1$ we have*

$$\Pr[X \geq \lambda\mu] \leq \lambda^{-1}(1 - p_*)^{(\lambda-1-\ln(\lambda))\mu}.$$

Corollary 12.3.5. *For $i \in [n]$ assume $X_i \sim \text{Geo}(p_i)$ for $p_i \in (0, 1]$. Let $X = \sum_{i \in [n]} X_i$ and write $\mu = \mathbb{E}[X]$, $p_* = \min_{i \in [n]} p_i$. Let $\rho \in (0, 1)$. Then $\Pr[X \geq T] \leq \rho$ whenever*

$$T \geq 2 \ln(2)\mu + 2 \frac{\ln(1/\rho)}{\ln(1/(1 - p_*))}.$$

Proof. We apply Lemma 12.3.4 with $\lambda \geq 1$ to obtain

$$\Pr[X \geq \lambda\mu] \leq \lambda^{-1}(1 - p_*)^{(\lambda-1-\ln(\lambda))\mu} \leq (1 - p_*)^{(\lambda-1-\ln(\lambda))\mu}.$$

By the first-order characterization of convexity of $\lambda \mapsto \lambda - 1 - \ln(\lambda)$ at $\lambda = 2$, we have

$$\lambda - 1 - \ln(\lambda) \geq \left(1 - \frac{1}{2}\right)(\lambda - 2) + 2 - 1 - \ln(2) = \frac{1}{2}\lambda - \ln(2).$$

Therefore

$$\Pr[X \geq \lambda\mu] \leq e^{\ln(1-p_*)(\lambda/2-\ln(2))\mu},$$

and so to ensure that this is at most ρ , it suffices to take

$$\lambda\mu \geq 2 \ln(2)\mu + \frac{2 \ln(\rho)}{\ln(1 - p_*)}.$$

Note that such λ also satisfies $\lambda \geq 1$, because $2 \ln(2) \geq 1$ and $\ln(\rho)/\ln(1 - p_*) \geq 0$. Therefore we have shown that $\Pr[X \geq T] \leq \rho$ whenever $T \geq 2 \ln(2)\mu + 2 \frac{\ln(\rho)}{\ln(1-p_*)}$. \square

Applying the above tail bound with $p_i \geq \frac{2}{3}(k - i + 1)/k$ yields the following lemma.

Lemma 12.3.6. *Let $1 \leq t \leq k \leq N$ and subset $I \subseteq [N]$ of size k , and let $\rho \in (0, 1)$. Consider a procedure in which at each step with probability $\geq 2/3$, one obtains a uniformly random sample from I . The outputs of*

$$r \geq 3 \ln(2) k(H_k - H_{k-t}) + \frac{2 \ln(1/\rho)}{\ln(3k/(k + 2(t-1)))} =: R_{t,k,\rho}$$

repetitions of this procedure suffice to, with probability $\geq 1 - \rho$, obtain t distinct samples from I .

We briefly emphasize the value of this lemma. For general $t \leq k$, we can use the simple bound $\ln(k/(t-1)) \geq \ln(k/(k-1)) \geq 1/k$ and the estimate $H_k - H_{k-t} \approx \ln(k/(k-t))$, to obtain that $r \in \Omega(k \log(k) + k \ln(1/\rho)) = \Omega(k \log(k/\rho))$ samples suffice. By contrast, an application of Markov's inequality only yields a sample complexity upper bound of $O(k \log(k) \log(1/\rho))$. In later applications (cf. Theorem 12.3.9), we apply this with t at most $k/2$, in which case we can give tighter estimates. Indeed, the factor $1/\ln(3k/(k + 2(t-1)))$ is then at most a constant and $H_k - H_{k-t} \leq \frac{2(t+1)}{k}$ by Lemma 12.3.3, and thus $r \in \Omega(t + \ln(1/\rho))$ samples suffice. Therefore the bound is an improvement over the sample complexity of $\Omega(t \ln(1/\rho))$ one would obtain from a simple application of Markov's inequality – in particular, one can now “for free” choose ρ to be exponentially small in t (and similar above).

By using $\text{Grover}_{2/3}$ to obtain the samples required for Lemma 12.3.6, we obtain the following algorithmic result.

Proposition 12.3.7. *Let $x \in \{0, 1\}^N$ with $|x| = k$ unknown, let $R \geq 1$, let $k_{\text{lb}} \geq 1$ be such that $k_{\text{lb}} \leq k$, let $t \geq 1$, and $\rho \in (0, 1)$. Assume $1 \leq t \leq k$. Then GroverCoupon called with a quantum oracle U_x to access x , and additional inputs R , k_{lb} , and t , uses*

Subroutine GroverCoupon(U_x, R, k_{lb}, t)

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$, an integer $R \geq 1$, an integer k_{lb} such that $k_{lb} \leq |x|$, an integer $t \geq 1$ such that $1 \leq t \leq |x|$.

Output: Classical sorted list of indices $J \subseteq [N]$.

Guarantee: If $R \geq R_{t,k,\rho} = 3 \ln(2)k(H_k - H_{k-t}) + 2 \frac{\ln(1/\rho)}{\ln(3k/(k+2(t-1)))}$, then, with probability $\geq 1 - \rho$, we have $|J| = t$ and $x_j = 1$ for all $j \in J$.

Analysis: Proposition 12.3.7

```

1  J ← ∅;
2  for r = 1, ..., R do
3      use Grover2/3 with arguments (Ux, klb) to find a j ∈ [N] such that xj = 1
         with probability ≥ 2/3 ;
4      if j ∉ J and xj = 1 then
5          | add j to J;
6      end if
7      if |J| = t then
8          | return J ;
9      end if
10 end for
11 return J;
```

$O(\sqrt{N/k_{lb}} r)$ quantum queries to x and $O(\sqrt{N/k_{lb}} r \log(N))$ additional quantum gates. Here, r is a random variable such that $r \leq R$ with certainty, and with probability $\geq 1 - \rho$, one has

$$r \leq R_{t,k,\rho} = 3 \ln(2)k(H_k - H_{k-t}) + 2 \frac{\ln(1/\rho)}{\ln(3k/(k+2(t-1)))}.$$

If $R \geq R_{t,k,\rho}$, then with probability $\geq 1 - \rho$, it finds a set of t distinct marked elements, uniformly at random from the set of all sets of k marked elements.

Proof. We first analyze the complexity of GroverCoupon. Let $r \in [R]$ be the number of times the algorithm repeats Line 3 through Line 8. By Lemma 11.2.5, the application of Grover_{2/3} in Line 3 uses $O(\sqrt{N/k_{lb}})$ quantum queries and $O(\sqrt{N/k_{lb}} \log(N))$ additional gates. With one additional query we can verify if the index $j \in [N]$ that is returned by Grover_{2/3} is such that $x_j = 1$. If indeed $x_j = 1$, then we add j to J . As mentioned in Section 12.2.2, we can insert an element in the sorted list J in (classical) time $O(\log(N))$. We can verify Line 7 in time $O(\log(N))$ by maintaining a counter for $|J|$. The above shows that GroverCoupon indeed uses $O(\sqrt{N/k_{lb}} r)$ quantum queries and $O(\sqrt{N/k_{lb}} r \log(N))$ additional quantum gates.

We now establish correctness. By construction $r \leq R$ with certainty. Lemma 11.2.5 shows that, with probability $\geq 2/3$, the index returned by Grover_{2/3} in Line 3 is a uniformly random marked element. Hence Lemma 12.3.6 shows that after obtaining

$$R_{t,k,\rho} = 3 \ln(2) k(H_k - H_{k-t}) + \frac{2 \ln(1/\rho)}{\ln(3k/(k+2(t-1)))}$$

such indices, we have obtained t distinct indices with probability at least $1 - \rho$. In other words, if $R \geq R_{t,k,\rho}$, then, with probability at least $1 - \rho$, GroverCoupon terminates at Line 8 with a sorted list $J \subseteq [N]$ of t distinct marked indices. \square

12.3.3. Grover for multiple elements, fast

In this section we improve the complexity of finding all marked indices by combining the two previously discussed algorithms, GroverCoupon and GroverCertaintyMultiple. The structure of our algorithm, GroverMultipleFast, is as follows. As before, suppose we are given query access to an $x \in \{0, 1\}^N$. Let the (unknown) number of marked indices be $k \geq 1$, i.e., $k = |x|$. We first use GroverCoupon to find a (large) fraction of the marked elements. That is, we find a uniformly random subset $J_0 \subseteq [N]$ of τk marked elements, where $0 < \tau < 1$ is a parameter we can use to tune the complexity of the algorithm. This subset J_0 partitions $[N]$ into intervals. We then use GroverCertaintyMultiple to find all marked indices in each interval separately.

The following lemma upper bounds the probability that when we draw a set $S \subseteq [k]$ of size t uniformly at random, there exists an interval of length $\geq \ell$ in the set $[k] \setminus S$. In the analysis of GroverMultipleFast (see Theorem 12.3.9), we will use this bound to control the number of elements that are in between any two elements of the previously sampled indices J_0 .

Lemma 12.3.8. *Let $S \subseteq [k]$ be a uniformly random t -element set, and let $1 \leq \ell \leq k - t$. The probability that $[k] \setminus S$ contains a contiguous subset I of length $\geq \ell$, i.e., $I = \{a, a + 1, \dots, a + \ell - 1\}$ for $1 \leq a \leq k - \ell + 1$, is at most $(k - \ell + 1)(1 - \frac{t}{k})^\ell$.*

Proof. The probability that $[k] \setminus S$ contains a contiguous subset I of length at least ℓ is the same as the probability that it contains a contiguous subset of length exactly ℓ . This is in turn given by

$$\Pr[\exists a \in \{1, \dots, k - \ell + 1\} : \{a, \dots, a + \ell - 1\} \cap S = \emptyset].$$

By a union bound, this is at most

$$\sum_{a=1}^{k-\ell+1} \Pr[\{a, \dots, a + \ell - 1\} \cap S = \emptyset].$$

By uniform randomness of S , each of these probabilities is the same, and given by

$$\Pr[\{a, \dots, a + \ell - 1\} \cap S = \emptyset] = \frac{\binom{k-\ell}{t}}{\binom{k}{t}} = \frac{(k-t)(k-t-1)\cdots(k-t-\ell+1)}{k(k-1)\cdots(k-\ell+1)}.$$

This is at most $(1 - t/k)^\ell$, and we conclude that the probability that $[k] \setminus S$ contains a contiguous subset I of at least ℓ is at most $(k - \ell + 1)(1 - \frac{t}{k})^\ell$. \square

Theorem 12.3.9. *Let $x \in \{0, 1\}^N$ with $|x| = k \geq 2$, and assume one knows $k_{\text{est}} \geq 1$ such that $k/2 \leq k_{\text{est}} \leq 3k/2$. Let $0 < \rho < 1$ and $6 \leq \lambda \leq k_{\text{est}}$ be such that $t := \lceil k_{\text{est}}/\lambda \rceil \geq \log_2(6k_{\text{est}}/\rho)$. Then*

$$O\left(\sqrt{Nk} \left(1 + \frac{1}{\sqrt{\lambda}} \log(k/\rho\lambda)\right)\right)$$

quantum queries to x suffice to, with probability $\geq 1 - \rho$, find all k indices i s.t. $x_i = 1$. The algorithm uses an additional

$$O\left(\sqrt{Nk}\lambda \log(k/\rho) \log(N)\right)$$

non-query gates.

Subroutine GroverMultipleFast($U_x, k_{\text{est}}, \rho, \lambda$)

Input: Quantum oracle U_x to access $x \in \{0, 1\}^N$, an integer $k_{\text{est}} \geq 1$ such that $|x|/2 \leq k_{\text{est}} \leq 3|x|/2$, a failure probability $\rho > 0$, threshold parameter $\lambda \in [6, k_{\text{est}}]$.

Output: Classical list of indices $J \subseteq [N]$.

Guarantee: If λ and ρ are such that $\log_2(6k_{\text{est}}/\rho) \leq \lceil k_{\text{est}}/\lambda \rceil$, then, with probability $\geq 1 - \rho$, we have $|J| = |x|$ and $x_j = 1$ for all $j \in J$.

Analysis: Theorem 12.3.9

```

1 J ← ∅;
2 t ← ⌈k/λ⌉;
3 R ← 6 ln(2)(t + 1) + 2 ln(1/ρ) ln(3/2);
4 use GroverCoupon( $U_x, R, \frac{2}{3}k_{\text{est}}, t$ ) to find, with probability  $\geq 1 - \rho/3$ , a
   sorted list  $J_0 \subseteq [N]$  with  $x_j = 1$  for all  $j \in J_0$ ,  $|J_0| = t$ ;
5 set  $J \leftarrow J_0$  and write  $J_0 = \{a_1 < a_2 < \dots < a_t\}$ ;
6 set  $a_0 = 0$  and  $a_{t+1} = N + 1$ ;
7 for  $i = 0, \dots, t$  do
8   If  $a_{i+1} = a_i + 1$ , continue with next loop; otherwise, let
    $b_i = 2^{\lceil \log_2(a_{i+1} - 1 - a_i) \rceil}$ ;
9   construct from  $U_x$  an oracle  $U_y$  which implements access to the bit string
    $y \in \{0, 1\}^{b_i}$  given by  $y_j = x_{a_i+j}$  if  $a_i + j < a_{i+1}$ , and 0 otherwise;
10   $(k_j)_{\text{est}} \leftarrow \text{ApproxCount}(U_y, \frac{1}{2}, \frac{\rho}{3(t+1)})$ ;
11  use GroverCertaintyMultiple( $U_y, 2(k_j)_{\text{est}}$ ) to find all  $j \in (a_i, a_{i+1})$  such
   that  $x_j = 1$ , and add these to  $J$ ;
12 end for
13 return J;
```

We remark here that GroverMultipleFast takes a multiplicative estimate k_{est} of k as additional input, which can be found with $O(\sqrt{N/k} \log(1/\rho))$ quantum queries and $O(\sqrt{N/k} \log(1/\rho) \log(N))$ additional gates; see Corollary 11.2.8. Both of these costs are dominated by that of finding the actual elements. The above theorem also includes a parameter λ that allows for a trade-off between query complexity and gate complexity. Before we provide the proof of Theorem 12.3.9, let us highlight the two extremal cases that follow from taking λ either as large as useful or as small as possible.

Corollary 12.3.10. *Let $x \in \{0, 1\}^N$ with $|x| = k \geq 2$. Assume one knows k_{est} such that $k/2 \leq k_{\text{est}} \leq 3k/2$. Let $1 > \rho > 0$. Then we can find, with probability $\geq 1 - \rho$, all k indices i for which $x_i = 1$ using either:*

- $O(\sqrt{Nk})$ quantum queries and time complexity $O(\sqrt{Nk} \min\{\log^3(k/\rho), k\} \log(N))$, via Theorem 12.3.9 with $\lambda = \min\{k_{\text{est}}/\log_2(6k_{\text{est}}/\rho), \log_2^2(k_{\text{est}}/\rho)\}$,⁴ or,
- $O(\sqrt{Nk} \log(k/\rho))$ quantum queries and time complexity $O(\sqrt{Nk} \log(k/\rho) \log(N))$, via Theorem 12.3.9 with $\lambda = 6$.

⁴Strictly speaking, this choice of λ could be smaller than 6, but in that case GroverCertaintyMultiple already has the stated complexity.

Proof of Theorem 12.3.9. Let $t = \lceil k_{\text{est}}/\lambda \rceil$. Note that because $\lambda \geq 6$ and $k_{\text{est}} \leq 3k/2$, we have $t \leq k/2$. Therefore we can find t of the solutions using the procedure of Proposition 12.3.7 with probability $\geq 1 - \rho/3$, using

$$O\left(\sqrt{\frac{N}{k}}(t + \log(1/\rho))\right) = O\left(\sqrt{\frac{N}{k}}\left(\frac{k}{\lambda} + \log(1/\rho)\right)\right) \quad (12.3.2)$$

queries and

$$O\left(\sqrt{\frac{N}{k}}\left(\frac{k}{\lambda} + \log(1/\rho)\right)\log(N)\right) \quad (12.3.3)$$

gates. We remark here that these upper bounds hold because $t \leq k/2 < k$. Indeed, under that assumption on t and k we have $k(H_k - H_{k-t}) \leq 2(t+1)$ by Lemma 12.3.3, and moreover the factor $1/\ln(3k/(k+2(t-1)))$ is $\Theta(1)$ (it lies between $1/\ln(3)$ and $1/\ln(3/2)$). This shows that calling `GroverCoupon` with $R = 6\ln(2)(t+1) + 2\ln(1/\rho)\ln(3/2) \in \Theta(t + \log(1/\rho))$ has the desired behaviour.

Let $a_1 < a_2 < \dots < a_t$ denote the found indices for which $x_{a_j} = 1$ and define the intervals $I_0 = \{1, \dots, a_1 - 1\}$, $I_t = \{a_t + 1, \dots, N\}$, and, for $j \in [t-1]$, $I_j = \{a_j + 1, \dots, a_{j+1} - 1\}$. We use k_j to denote the (unknown) number of marked elements in I_j , so in particular $\sum_{j=0}^t k_j \leq k - t$. Then by Lemma 12.3.8, the probability that there is a k_j larger than $\ell := \frac{k}{t}(\log_2(k) + \log_2(3/\rho))$ is at most

$$(k - \ell + 1) \left(1 - \frac{t}{k}\right)^\ell \leq 2^{\log_2(k)} \left(1 - \frac{t}{k}\right)^{\frac{k}{t}(\log_2(k) + \log_2(3/\rho))} \leq k \left(\frac{1}{2}\right)^{\log_2(k) + \log_2(3/\rho)} = \rho/3.$$

Here we used that $\ell \geq 1$, $(1 - \frac{t}{k})^{k/t} \leq \frac{1}{e} \leq \frac{1}{2}$, and $\log_2(k) + \log_2(3/\rho) \geq 0.5$. For the rest of the argument we may thus assume that there is no interval with more than ℓ not-yet-found marked elements.

In the next step of our algorithm we search for all marked elements in each interval. To do so for the j th interval, we search over the elements from $[2^{\lceil \log_2(|I_j|) \rceil}]$ marking an element $i \in [2^{\lceil \log_2(|I_j|) \rceil}]$ if $x_{i+a_j} = 1$ and $i \leq |I_j|$ (letting $a_0 = 0$). One can implement this unitary using $O(1)$ quantum queries and $O(\log(N))$ gates (to implement the addition and comparison). For each interval, we first compute an estimate $(k_j)_{\text{est}}$ of k_j that satisfies $k_j/2 \leq (k_j)_{\text{est}} \leq 3k_j/2$ using Corollary 11.2.8, with success probability $\geq 1 - \rho/(3(t+1))$. The associated query cost is $O(\sqrt{|I_j|/(k_j+1)}\log(t/\rho))$, and it uses $O(\sqrt{|I_j|/(k_j+1)}\log(t/\rho)\log(N))$ additional gates. Then Lemma 12.3.2 shows that we can find all marked elements in the j -th interval with probability 1 using $O(\sqrt{|I_j|}(k_j)_{\text{est}})$ quantum queries and $O(\sqrt{|I_j|}(k_j)_{\text{est}}^{3/2}\log(N))$ additional gates. By a union bound, with probability $\geq 1 - \rho/3$, all $(k_j)_{\text{est}}$ are correct, and this step has a total query complexity of

$$O\left(\sum_{j=0}^t \sqrt{|I_j|k_j} + \sqrt{|I_j|}\log(t/\rho)\right) = O\left(\sqrt{Nk} + \sqrt{Nt}\log(t/\rho)\right)$$

⁵Note also that $\ell \leq k$ because $t \geq \log_2(6k_{\text{est}}/\rho) \geq \log_2(3k/\rho)$ by assumption; if $\ell > k$, then the probability of having an interval of length $\geq \ell$ is of course 0, and in this regime one may just as well run `GroverCertaintyMultiple` on the whole string (and have zero failure probability).

$$= O\left(\sqrt{Nk}\left(1 + \frac{\log(k/\rho\lambda)}{\sqrt{\lambda}}\right)\right), \quad (12.3.4)$$

where the first step uses Cauchy–Schwarz for both terms (reading $\sqrt{|I_j|}$ as $\sqrt{|I_j|} \cdot 1$ for the second term) and $\sum_{j=0}^t |I_j| \leq N$, $\sum_{j=0}^t k_j = k$. To analyze the gate complexity of this step, we first bound $\sum_{j=0}^t k_j^3$. We have $\|\mathbf{k}^2\|_\infty \leq \ell^2 = O(\lambda^2 \log^2(3k/\rho))$ where \mathbf{k} is the vector with entries k_j and \mathbf{k}^2 is the entrywise square of \mathbf{k} . As we also have $\|\mathbf{k}\|_1 \leq k$ we get $\sum_{j=0}^t k_j^3 = \langle \mathbf{k}, \mathbf{k}^2 \rangle \leq \|\mathbf{k}\|_1 \|\mathbf{k}^2\|_\infty = O(k\ell^2)$. Then the gate complexity of the final search steps becomes:

$$\begin{aligned} & O\left(\left(\sum_{j=0}^t \sqrt{|I_j|} k_j^3 + \sum_{j=0}^t \sqrt{|I_j|} \log(t/\rho)\right) \log(N)\right) \\ &= O\left(\sqrt{N} \left(\sqrt{\sum_{j=0}^t k_j^3} + \sqrt{t} \log(t/\rho)\right) \log(N)\right) \\ &= O\left(\sqrt{N} \left(\sqrt{k\ell} + 1 + \sqrt{t} \log(t/\rho)\right) \log(N)\right) \\ &= O\left(\sqrt{N} \left(\sqrt{k\lambda} \log(k/\rho) + \sqrt{k/\lambda} \log(k/\rho\lambda)\right) \log(N)\right) \\ &= O\left(\sqrt{Nk} \left(\lambda \log(k/\rho) + \sqrt{1/\lambda} \log(k/\rho\lambda)\right) \log(N)\right) \\ &= O\left(\sqrt{Nk\lambda} \log(k/\rho) \log(N)\right), \end{aligned} \quad (12.3.5)$$

where we again used Cauchy–Schwarz in the first step, and the total error probability is bounded by $\rho/3 + \rho/3 + (t+1) \cdot \frac{\rho}{3(t+1)} = \rho$.

To conclude, the upper bound on the total query complexity follows by combining Eqs. (12.3.2) and (12.3.4):

$$\begin{aligned} & O\left(\underbrace{\sqrt{\frac{N}{k}} \left(\frac{k}{\lambda} + \log(1/\rho)\right)}_{\text{sample } t \text{ elements}} + \underbrace{\sqrt{Nk} \left(1 + \frac{1}{\sqrt{\lambda}} \log(k/\rho\lambda)\right)}_{\text{find remaining elements}}\right) \\ &= O\left(\sqrt{Nk} \left(1 + \frac{1}{\sqrt{\lambda}} \log(k/\rho\lambda)\right) + \sqrt{\frac{N}{k}} \log(1/\rho)\right) = O\left(\sqrt{Nk} \left(1 + \frac{1}{\sqrt{\lambda}} \log(k/\rho\lambda)\right)\right). \end{aligned}$$

Here the first equality uses that $\sqrt{\frac{N}{k} \frac{k}{\lambda}} \leq \sqrt{Nk}$ since $\lambda \geq 1$. The second equality follows since $\log_2(1/\rho) \leq \log_2(6k_{\text{est}}/\rho)$ and, by assumption, $\log_2(6k_{\text{est}}/\rho) \leq \lceil k_{\text{est}}/\lambda \rceil = t \leq k$. A similar argument using Eqs. (12.3.3) and (12.3.5) and $\lambda \geq 1$, establishes the desired gate complexity:

$$\begin{aligned} & O\left(\underbrace{\sqrt{\frac{N}{k}} \left(\frac{k}{\lambda} + \log(1/\rho)\right) \log(N)}_{\text{sample } t \text{ elements}} + \underbrace{\sqrt{Nk\lambda} \log(k/\rho) \log(N)}_{\text{find remaining elements}}\right) \\ &= O\left(\sqrt{Nk} \left(\frac{1}{\lambda} + \lambda \log(k/\rho)\right) \log(N) + \sqrt{\frac{N}{k}} \log(1/\rho) \log(N)\right) \end{aligned}$$

$$= O\left(\sqrt{Nk\lambda} \log(k/\rho) \log(N)\right). \quad \square$$

12.4. Improved query complexity for approximate summation

In this section, we provide an algorithm `ApproxSum`, which given quantum query access to a binary description of $v \in [0, 1]^N$, in the sense of Definition 12.2.2, finds a $(1 \pm \delta)$ -multiplicative approximation of $s = \sum_{i=1}^N v_i$ with probability $\geq 1 - \rho$ using

$$O\left(\sqrt{\frac{N}{\delta} \log(1/\rho)}\right) \quad (12.4.1)$$

quantum queries and a similar gate complexity (with only a polylogarithmic overhead). In the above (12.4.1) we have made very mild assumptions on the value of ρ and δ ; a precise statement is given in Theorem 12.4.3. The algorithm is given in `ApproxSum`. By slightly perturbing the entries of v , we may assume without loss of generality that all entries of v are distinct; we shall make this assumption throughout this section, and have made this assumption in the description of the algorithm as well.

We briefly explain the overall strategy. Recall from the proof of Theorem 11.2.7 that it is useful to preprocess the vector v by using quantum maximum finding to find $v_{\max} = \max_{i \in [N]} v_i$, and then to use amplitude estimation on the vector $w = v/v_{\max}$. We take this approach slightly further: we first find the largest k entries z_1, \dots, z_k of v , where $k = \Theta(pN)$ for $p \in (0, 1)$, and sum their values classically. Let \tilde{z} be the smallest value among the z_1, \dots, z_k .⁶ For the next part, we treat the corresponding entries of v as zero: checking whether one exceeds the threshold \tilde{z} is a binary comparison, hence can be done in superposition without explicitly using their indices, and so with one query to v we can implement quantum oracle access to the vector $w \in [0, 1]^N$ defined by

$$w_i = \begin{cases} \frac{v_i}{\tilde{z}} & \text{if } v_i < \tilde{z} \\ 0 & \text{else.} \end{cases}$$

This has the effect of amplifying the small elements in v at no extra cost. We then use amplitude estimation to compute $\sum_{i=1}^N w_i$ with additive precision $O(\delta s/\tilde{z})$ (without knowing s). This yields an additive δs -approximation of $\sum_{i=1}^N v_i$ (i.e., a $(1 \pm \delta)$ -multiplicative approximation), where we use that

$$\sum_{i=1}^N v_i = \sum_{i=1}^k z_i + \tilde{z} \sum_{i=1}^N w_i$$

To balance the costs of these two stages we need to carefully choose \tilde{z} . We do so by estimating the p -th quantile of the vector. We first give an algorithm `ApproxSum`

⁶We actually first compute a good value of \tilde{z} using a quantile estimation subroutine [Ham21, Thm. 3.4] and then find all the z_j 's. Alternatively, one could use [DHHM06, Thm. 3.4] to find all $\Theta(pN)$ largest elements directly, but our approach has the advantage of being able to use the better ρ -dependence of our version of Grover search.

Subroutine ApproxSum($U_v, \delta, p, \lambda, \rho$)

Input: Quantum query access U_v to $(0, b)$ -fixed point representations of $v \in [0, 1]^N$, $\delta \in (0, 1)$, $p \in (0, 1)$, $\lambda \geq 6$, failure probability $\rho > 0$.

Output: A real number \tilde{s} .

Guarantee: With probability $\geq 1 - \rho$, \tilde{s} is a $(1 \pm \delta)$ -multiplicative approximation of s .

Analysis: Theorem 12.4.3

- 1 use Theorem 12.4.2 to compute $\tilde{z} \in [0, 1]$ such that with probability $\geq 1 - \rho/4$, $Q(p) \leq \tilde{z} \leq Q(cp)$, where $c < 1$ is a universal constant and Q is defined in Eq. (12.4.2);
 - 2 let $x \in \{0, 1\}^N$ be defined by $x_i = 1$ if $v_i \geq \tilde{z}$ and $x_i = 0$ otherwise;
 - 3 let U_x implement quantum query access to x by applying U_v , comparing to \tilde{z} , and uncomputing U_v ;
 - 4 compute estimate k_{est} of $k = |x|$ satisfying $\frac{k}{2} \leq k_{\text{est}} \leq \frac{3k}{2}$ with probability $\geq 1 - \rho/4$ using Corollary 11.2.8;
 - 5 use GroverMultipleFast($U_x, k_{\text{est}}, \rho/4, \lambda$) to find all indices i_1, \dots, i_k such that $x_{i_j} = 1$;
 - 6 **if** $\tilde{z} = 0$ **then**
 - 7 | **return** $\sum_{j=1}^k v_{i_j}$;
 - 8 **else**
 - 9 | construct unitary U_w for query access to $w \in [0, 1]^N$ where $w_i = 0$ if $v_i \geq \tilde{z}$ and $w_i = v_i/\tilde{z}$ otherwise;
 - 10 | let U be a unitary such that $U|0\rangle = |\psi\rangle$ given by

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_i |i\rangle (\sqrt{\tilde{w}_i} |1\rangle + \sqrt{1 - \tilde{w}_i} |0\rangle)$$
 where α_i is a $\lceil \log_2(4N/\delta) \rceil$ -bit approximation of $\arcsin(\sqrt{w_i})$, and $\sqrt{\tilde{w}_i} = \sin(\alpha_i)$;
 - 11 | use AmpEst(U, M) with $M = \lceil 12\pi\sqrt{\delta^2 pc} \rceil$, increased to the next power of 2 if necessary, with $c < 1$ from Theorem 12.4.2, to compute $\tilde{a} \approx \sum_i \tilde{w}_i/N$, and repeat $O(\log(1/\rho))$ times and take the mean of the outputs to achieve success probability $\geq 1 - \rho/4$;
 - 12 | **return** $\sum_{j=1}^k v_{i_j} + N\tilde{z}\tilde{a}$;
 - 13 **end if**
-

whose complexity depends on the quantile p and then give a suitable choice for p that allows us to obtain (12.4.1), see Theorem 12.4.3 and Corollary 12.4.4.

We use the following lemma to derive a bound on the required precision for certain arithmetic operations.

Lemma 12.4.1 ([BHMT02, Lem. 7]). *If $a = \sin^2(\theta_a)$ and $\tilde{a} = \sin^2(\tilde{\theta}_a)$ for $\theta_a, \tilde{\theta}_a \in [0, 2\pi]$, then $|\tilde{\theta}_a - \theta_a| \leq \delta$ implies $|\tilde{a} - a| \leq 2\delta\sqrt{a(1-a)} + \delta^2$.*

For the quantile estimation, we use a subroutine from [Ham21]. Let $v \in [0, 1]^N$. Then for $p \in (0, 1)$, we define the p -quantile $Q(p) \in [0, 1]$ by

$$Q(p) = \sup\{z \in [0, 1] : |\{i \in [N] : v_i \geq z\}| \geq pN\}. \quad (12.4.2)$$

In words, $Q(p)$ is the largest value $z \in [0, 1]$ such that there are at least pN entries of v which are larger than z . The subroutine we invoke allows one to produce an estimate for $Q(p)$, in the following sense:

Theorem 12.4.2 ([Ham21, Thm. 3.4]). *There exists a universal constant $c \in (0, 1)$ such that the following holds: Let $v \in [0, 1]^N$ and let U_v be a unitary implementing quantum oracle access to v . Then $O(\log(1/\rho)/\sqrt{p})$ applications of controlled- U_v and controlled- U_v^\dagger suffice to find, with probability $\geq 1 - \rho$, a value \tilde{z} such that $Q(p) \leq \tilde{z} \leq Q(cp)$. The algorithm uses an additional $O((\log(1/\rho)/\sqrt{p}) b \log(b) \log(N))$ gates.*

The actual access model for which the above theorem holds is more general, but we have instantiated it for our setting. The gate complexity overhead follows from having to implement their access model using ours, which involves arithmetic and comparisons on the fixed point representations we use, and the fact that the underlying technique is amplitude amplification. We now get to the main theorem of this section, which proves the correctness of `ApproxSum` and analyzes its complexity.

Theorem 12.4.3. *Let $v \in [0, 1]^N$, let U_v be a unitary implementing quantum query access to $(0, b)$ -fixed point representations of v , and let $\delta \in (0, 1)$. Let $p, \rho \in (0, 1)$ and choose $6 \leq \lambda \leq \min\{cpN/\log_2(pN/\rho), \log_2(cpN/\rho)^2\}$. Then `ApproxSum` computes, with probability $\geq 1 - \rho$, a $(1 \pm \delta)$ -multiplicative approximation of $s = \sum_{i=1}^N v_i$. It uses*

$$O\left(\frac{\log(1/\rho)}{\sqrt{p}} + \sqrt{\frac{N}{Np+1}} \log(1/\rho) + N\sqrt{p} \left(1 + \frac{1}{\sqrt{\lambda}} \log(Np/\lambda\rho)\right) + \frac{1}{\delta\sqrt{p}} \log(1/\rho)\right)$$

quantum queries, and the number of additional gates is bounded by

$$O\left(\frac{\log(1/\rho)}{\sqrt{p}} b \log(b) \log(N) + \sqrt{\frac{N}{Np+1}} \log(1/\rho) \log(N) + N\sqrt{p}\lambda \log(pN/\rho) \log(N) + \frac{1}{\delta\sqrt{p}} b \log(b) \log(N/\delta) \log^2 \log(N/\delta) \log(1/\rho)\right).$$

Before we give the proof, we discuss two useful regimes for p and λ :

Corollary 12.4.4. *Let $v \in [0, 1]^N$, let U_v be a unitary implementing quantum oracle access to $(0, b)$ -fixed point representations of v , and let $\delta \in (0, 1)$. Then we can find, with probability $\geq 1 - \rho$, a $(1 \pm \delta)$ -multiplicative approximation of $s = \sum_{i=1}^N v_i$, using:*

- $O(\sqrt{N \log(1/\rho)/\delta})$ quantum queries, when $p = \Theta(\log(1/\rho)/(\delta N)) < 1$ and we choose $\lambda = \min\{cpN/\log_2(6pN/\rho), \log_2(cpN/\rho)^2\} \geq 6$, and using $O(\sqrt{N/\delta} \text{poly}(\log(1/\rho), b, \log(N), \log(1/\delta)))$ additional gates, or
- $O(\sqrt{N/\delta} \log(1/\rho))$ quantum queries when $p = \Theta(1/(\delta N)) < 1$ and we choose $\lambda = 6$, and using $\sqrt{N/\delta} \text{poly}(\log(1/\rho), b, \log(N), \log(1/\delta))$ additional gates.

Proof of Theorem 12.4.3. We assume without loss of generality that all the entries of v are distinct. If this is not the case, one can perturb the i -th entry of v by $i2^{-\ell}$ for some sufficiently large $\ell = \Omega(\log(N) + b)$, where we recall that b is the number of bits describing v_i , and discarding these trailing bits from the output value \tilde{s} .

12. Basic quantum subroutines, improved

We use Theorem 12.4.2 to find a value \tilde{z} such that the number of elements of v that are at least as large as \tilde{z} , is at most pN and at least cpN . The number of quantum queries is

$$O\left(\frac{\log(1/\rho)}{\sqrt{p}}\right),$$

and the number of additional gates used is

$$O\left(\frac{\log(1/\rho)}{\sqrt{p}} b \log(b) \log(N)\right).$$

Let $k = |\{i \in [N] : v_i \geq \tilde{z}\}|$. By the assumption that the v_i are all distinct, $cpN \leq k \leq pN$. We next compute a $(1 \pm \frac{1}{2})$ -multiplicative approximation of k using Corollary 11.2.8. This uses

$$O\left(\sqrt{N/(k+1)} \log(1/\rho)\right)$$

quantum queries and

$$O\left(\sqrt{N/(k+1)} \log(1/\rho) \log(N)\right)$$

additional gates. The next step is to find all k such elements using GroverMultipleFast (Theorem 12.3.9). This uses

$$O\left(\sqrt{Nk} \left(1 + \frac{1}{\sqrt{\lambda}} \log(k/(\lambda\rho))\right)\right)$$

quantum queries and

$$O\left(\sqrt{Nk\lambda} \log(k/\rho) \log(N)\right)$$

additional gates.

Let z_1, \dots, z_k be the entries of v that are $\geq \tilde{z}$. Then

$$\sum_{i=1}^N v_i = \sum_{j=1}^k z_j + \tilde{z} \sum_{i=1}^N w_i$$

where

$$w_i = \begin{cases} \frac{v_i}{\tilde{z}} & \text{if } v_i < \tilde{z} \\ 0 & \text{otherwise.} \end{cases}$$

As we have found all the z_j 's, we can compute their sum exactly; therefore, to determine a $(1 \pm \delta)$ -multiplicative approximation of s , we must produce an additive δs -approximation of $\tilde{z} \sum_{i=1}^N w_i$. Let $\varepsilon := \delta s$; note that we do not know s as we do not know δ . Then we have to approximate $\frac{1}{N} \sum_{i=1}^N w_i$ with precision $\varepsilon/(N\tilde{z})$. For this, we use amplitude estimation as follows. First, one can implement query access to U_w by using two quantum queries to v and $O(b \log(b))$ non-query gates, by querying an entry, comparing the entry to \tilde{z} , and conditional on the comparison

uncomputing the query, and lastly performing the division by \tilde{z} . From this, we can construct a unitary U with $U|0\rangle = |\psi\rangle$ satisfying

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_i |i\rangle \left(\sqrt{\tilde{w}_i} |1\rangle + \sqrt{1 - \tilde{w}_i} |0\rangle \right),$$

where \tilde{w}_i is close to w_i . One can implement such a unitary as follows. First, set up a uniform superposition over the index register using $O(\log(N))$ gates. Use U_w to load binary descriptions of the entries of w . Calculate a $\lceil \log_2(4N/\delta) \rceil$ -bit approximation α_i of $\arcsin(\sqrt{w_i})$ using $O(\log(bN/\delta) \log^2 \log(bN/\delta))$ gates [BZ11, Ch. 4]. Then conditionally rotate the last qubit from 0 to 1 over angles $\pi/4, \pi/8$, et cetera, depending on the bits of α_i . Lastly, we uncompute α_i and U_w to return work registers to the zero state, and we have obtained the desired state $|\psi\rangle$, where $\sqrt{\tilde{w}_i} = \sin(\alpha_i)$. We now show that $\tilde{w}_i = \sin(\alpha_i)^2$ is close to w_i , and hence

$$a := \frac{1}{N} \sum_{i=1}^N \tilde{w}_i = \|\psi_1\|^2$$

is close to $\frac{1}{N} \sum_{i=1}^N w_i$. Lemma 12.4.1 shows that if $|\alpha_i - \arcsin(\sqrt{w_i})| \leq \xi$, then

$$|\tilde{w}_i - w_i| = |\sin^2(\alpha_i) - w_i| \leq 2\xi\sqrt{w_i(1-w_i)} + \xi^2 \leq \xi + \xi^2.$$

Since α_i is a $\lceil \log_2(4N/\delta) \rceil$ -bit approximation of $\arcsin(\sqrt{w_i})$, we may apply the above with $\xi = \delta/(4N)$ for every $i \in [N]$. Because $s \geq \tilde{z}$, $\delta = \varepsilon/s \leq \varepsilon/\tilde{z}$, and $\delta \leq 1$, so the total error satisfies

$$\left| a - \frac{1}{N} \sum_{i=1}^N w_i \right| \leq \frac{1}{N} \sum_{i=1}^N |\tilde{w}_i - w_i| = \xi + \xi^2 \leq \frac{\delta}{4N} + \frac{\delta^2}{16N^2} \leq \frac{\varepsilon}{2N\tilde{z}}.$$

Next, we use this to derive an upper bound on a :

$$a \leq \frac{\varepsilon}{2N\tilde{z}} + \frac{1}{N} \sum_{i=1}^N w_i = \frac{\varepsilon}{2N\tilde{z}} + \frac{1}{N} \sum_{i:v_i < \tilde{z}} \frac{v_i}{\tilde{z}} \leq \frac{2s}{N\tilde{z}}'$$

where the last inequality uses $\varepsilon = \delta s \leq s$ and $\sum_{i:v_i < \tilde{z}} v_i \leq s$. Therefore, using AmpEst with M applications of U yields a number $\tilde{a} \in [0, 1]$ with

$$|\tilde{a} - a| \leq 2\pi \frac{\sqrt{a(1-a)}}{M} + \frac{\pi^2}{M^2} \leq 2\pi \frac{\sqrt{2s/(N\tilde{z})}}{M} + \frac{\pi^2}{M^2}$$

by Theorem 11.2.2. We now determine an appropriate number of rounds M to be used for amplitude estimation. We will choose M such that $|\tilde{a} - a| \leq \frac{1}{2}\varepsilon/(N\tilde{z})$; if we do so, then by the triangle inequality $|\tilde{a} - \frac{1}{N} \sum_{i=1}^N w_i| \leq \varepsilon/(N\tilde{z})$. The claim is that any $M \geq 12\pi\sqrt{N\tilde{z}/(\varepsilon\delta)}$ suffices, as then

$$2\pi \frac{\sqrt{2s/(N\tilde{z})}}{M} \leq \frac{2\pi\sqrt{2}}{12\pi} \frac{\sqrt{s/(N\tilde{z})}}{\sqrt{N\tilde{z}/(\varepsilon\delta)}} = \frac{\sqrt{2}}{6} \frac{\varepsilon}{N\tilde{z}} \leq \frac{1}{4} \frac{\varepsilon}{N\tilde{z}}'$$

12. Basic quantum subroutines, improved

and, using $\delta \leq 1$,

$$\frac{\pi^2}{M^2} \leq \frac{\varepsilon \delta}{144N\tilde{z}} \leq \frac{1}{4} \frac{\varepsilon}{N\tilde{z}}.$$

Even though we do not know ε , by choosing p carefully, we can enforce upper bounds on \tilde{z} and give a safe choice for M . We use that the number of entries k which are at least \tilde{z} satisfies $k \geq cpN$, so that

$$cpN\tilde{z} \leq \sum_{i:v_i \geq \tilde{z}} v_i \leq s,$$

i.e., $\tilde{z} \leq s/(cpN)$. Therefore it suffices to take $M = 12\pi/\sqrt{\delta^2 pc}$, as this satisfies

$$M = 12\pi\sqrt{\frac{1}{\delta^2 pc}} = 12\pi\sqrt{\frac{s}{\delta \varepsilon pc}} \geq 12\pi\sqrt{\frac{N\tilde{z}}{\delta \varepsilon}},$$

This guarantees that $|\tilde{a} - \frac{1}{N} \sum_{i=1}^N w_i| \leq \varepsilon/(N\tilde{z})$, and the output value $\tilde{s} = \sum_{j=1}^k z_j + N\tilde{z}\tilde{a}$ satisfies

$$|\tilde{s} - s| \leq \varepsilon = \delta s.$$

The number of quantum queries used for this step is therefore $O(M) = O(1/(\delta\sqrt{p}))$, and the number of additional gates used is $O(Mb \log(b) \log(N/\delta) \log^2 \log(N/\delta))$. To amplify the success probability to $1 - \rho$, we repeat the above procedure $\log(1/\rho)$ many times and output the median of the individual estimates. The query- and gate complexity of the entire algorithm follow by combining those of the four parts: the quantile estimation, the approximate counting, Grover search for finding all large elements, and amplitude estimation for approximating the sum of the small elements. \square

13. Matrix scaling and matrix balancing

In this chapter, we provide a detailed introduction to the matrix scaling and matrix balancing problems, and the classical- and quantum state of the art for algorithms for solving these problems. It also serves as an overview of the results for Chapters 14 to 17.

13.1. Introduction

13.1.1. Matrix scaling and matrix balancing

Matrix scaling is a basic linear-algebraic problem with many applications. A *scaling* of an $n \times n$ matrix \mathbf{A} with non-negative entries is a matrix $\mathbf{B} = \mathbf{X}\mathbf{A}\mathbf{Y}$ where \mathbf{X} and \mathbf{Y} are positive diagonal matrices.¹ In other words, we multiply the i -th row with X_{ii} and the j -th column with Y_{jj} . We say \mathbf{A} is *exactly scalable* to marginals $\mathbf{r} \in \mathbb{R}_{>0}^n$ and $\mathbf{c} \in \mathbb{R}_{>0}^n$ if there exist \mathbf{X} and \mathbf{Y} such that the vector $\mathbf{r}(\mathbf{B}) = (\sum_{j=1}^n B_{ij})_{i \in [n]}$ of row sums of the scaled matrix \mathbf{B} equals \mathbf{r} , and its vector $\mathbf{c}(\mathbf{B})$ of column sums equals \mathbf{c} . We are given matrix \mathbf{A} and *target marginals* $\mathbf{r} \in \mathbb{R}_{>0}^n$ and $\mathbf{c} \in \mathbb{R}_{>0}^n$, and the goal is to find \mathbf{X} and \mathbf{Y} that yield those target marginals. One typical example would be if \mathbf{r} and \mathbf{c} are the all-1 vectors, which means we want $\mathbf{B} = \mathbf{X}\mathbf{A}\mathbf{Y}$ to be doubly stochastic: the rows and columns of \mathbf{B} would then be probability distributions.

In many cases it suffices to find *approximate* scalings. Different applications use different notions of approximation. We could for instance require $\mathbf{r}(\mathbf{B})$ to be ε -close to \mathbf{r} in ℓ^1 -norm or ℓ^2 -norm, or in relative entropy (Kullback-Leibler divergence), for some parameter ε of our choice, and similarly require $\mathbf{c}(\mathbf{B})$ to be close to \mathbf{c} .

A related problem is *matrix balancing*. Here we do not prescribe desired marginals, but the goal is to find a diagonal \mathbf{X} such that the row and column marginals of $\mathbf{B} = \mathbf{X}\mathbf{A}\mathbf{X}^{-1}$ are close to *each other*. Again, different quantitative notions of closeness $\mathbf{r}(\mathbf{B}) \approx \mathbf{c}(\mathbf{B})$ are possible.

An important application, used in theory as well as in practical linear-algebra software (e.g. LAPACK [ABB+99] and MATLAB [Mat]), is in improving the numerical stability of linear-system solving. Suppose we are given matrix \mathbf{A} and vector \mathbf{b} , and we want to find a solution to the linear system $\mathbf{A}\mathbf{v} = \mathbf{b}$. Note that \mathbf{v} is a solution iff $\mathbf{B}\mathbf{v}' = \mathbf{b}'$ for $\mathbf{v}' = \mathbf{X}\mathbf{v}$ and $\mathbf{b}' = \mathbf{X}\mathbf{b}$. An appropriately balanced matrix \mathbf{B} will typically be more numerically stable than the original \mathbf{A} , so solving the linear system $\mathbf{B}\mathbf{v}' = \mathbf{b}'$ and then computing $\mathbf{v} = \mathbf{X}^{-1}\mathbf{v}'$, is often a better way to solve the linear system $\mathbf{A}\mathbf{v} = \mathbf{b}$ than directly computing $\mathbf{A}^{-1}\mathbf{b}$.

This chapter is adapted from [AGL+21; GN22].

¹We assume \mathbf{A} is square for simplicity, but everything can straightforwardly be extended to non-square matrices.

13. Matrix scaling and matrix balancing

Matrix scaling and balancing have surprisingly many and wide-ranging applications. Matrix scaling was introduced by Kruithof for Dutch telephone traffic computation [Kru37], and has also been used in other areas of economics [Sto64]. In theoretical computer science it has been used for instance to approximate the permanent of a given matrix [LSW00], and for approximating optimal transport distances [ANR17]. In mathematics, it has been used as a common tool in practical linear algebra computations [LG04; Bra10; PC11; OCPB16], but also in statistics [Sin64], optimization [RS89], and for strengthening the Sylvester-Gallai theorem [BDWY11]. Matrix balancing has a similarly wide variety of applications, including pre-conditioning to make practical matrix computations more stable (as mentioned above), and approximating the min-mean-cycle in a weighted graph [AP22]. Many more applications of matrix scaling and balancing are mentioned in [LSW00; Ide16; GO18].

13.1.2. State of the art of classical algorithms

State of the art

Historically, research on matrix scaling and matrix balancing (and generalizations such as operator scaling) has focused on finding ε - ℓ^2 -scalings. More recently also algorithms for finding ε - ℓ^1 -scalings have been extensively studied, due to their close connection with permanents and finding perfect matchings in bipartite graphs [LSW00; CK21], and because the ℓ^1 -distance is an important error measure for statistical problems such as computing the optimal transport distance between distributions [Cut13; ANR17], even already for constant ε . By the Cauchy-Schwarz inequality, an (ε/\sqrt{n}) - ℓ^2 -scaling for \mathbf{A} is also an ε - ℓ^1 -scaling, but often more direct arguments can be given for the complexity of finding an ε - ℓ^1 -scaling.

Below in Table 13.1 we tabulate the best known algorithms for finding ε -scalings in ℓ^1 -norm for entrywise-positive matrices and general entrywise non-negative matrices, and we expand on this table later on.² For the well-definedness of the algorithms, we will always assume the $n \times n$ input matrix \mathbf{A} has at least one non-zero entry in every row and column, and every entry of the target marginals \mathbf{r}, \mathbf{c} is non-zero. In addition, we assume \mathbf{A} is *asymptotically scalable*: for every $\varepsilon > 0$, there exist \mathbf{X} and \mathbf{Y} such that

$$\|\mathbf{r}(\mathbf{B}) - \mathbf{r}\|_1 + \|\mathbf{c}(\mathbf{B}) - \mathbf{c}\|_1 \leq \varepsilon, \quad (13.1.1)$$

where $\mathbf{B} = \mathbf{XAY}$. A sufficient condition for this is that the matrix is entrywise-positive [Sin67]. As is standard in the matrix scaling literature, we will henceforth assume that \mathbf{A} is *asymptotically (\mathbf{r}, \mathbf{c}) -scalable*: for every $\varepsilon > 0$, there exist \mathbf{x}, \mathbf{y} such that $\mathbf{A}(\mathbf{x}, \mathbf{y})$ satisfies Eq. (13.1.1). This depends only on the support of \mathbf{A} [RS89, Thm. 3], and is the case if and only if (\mathbf{r}, \mathbf{c}) is in the convex hull of the points $(\mathbf{e}_i, \mathbf{e}_j) \in \mathbb{R}^{2n}$ such that $A_{ij} > 0$, where the \mathbf{e}_i are the standard basis vectors for \mathbb{R}^n . We will also always assume that the smallest non-zero entry of each of \mathbf{A}, \mathbf{r}

²For entrywise-positive matrices, the second-order methods (i.e., those that use the Hessian, not just the gradient) theoretically outperform the *classical* first-order methods in any parameter regime. However, they depend on highly non-trivial results for graph sparsification and Laplacian system solving which are relatively complicated to implement in practice, in contrast to the eminently practical Sinkhorn and Osborne algorithms.

and \mathbf{c} is at least $1/\text{poly}(n)$. To state the complexity results, let m be the number of non-zero entries in \mathbf{A} , assume $\sum_{i,j=1}^n A_{ij} = 1$, assume that its non-zero entries lie in $[\mu, 1]$, and $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$ (so a uniform marginal would be $1/n$). We will assume $\varepsilon \in (0, 1)$.³ The input numbers to the algorithm are all assumed to be rational, with bit size bounded by $\text{polylog}(n)$, unless specified otherwise.

	Time complexity	References and remarks
General non-negative	$\tilde{O}(m/\varepsilon^2)$	Sinkhorn, via KL [CK21] ⁴
	$\tilde{O}(mn/(h^{1/3}\varepsilon^{2/3}))$	first-order, via ℓ^2 [ALOW17]
	$\tilde{O}(m \log(\kappa))$	box-constrained, via ℓ^2 [CMTV17]
	$\tilde{O}(m^{1.5})$	interior-point method, via ℓ^2 [CMTV17]
	$\tilde{O}(m^{1+o(1)})$	[CKL+22; BCK+23], ε -dependence not explicit
	$\tilde{O}(\sqrt{mn}/\varepsilon^3)$	Sinkhorn, quantum, Corollary 14.1.9
	$\tilde{O}(\sqrt{mn} \log(\kappa)^{1.5}/\varepsilon)$	box-constrained, quantum, Corollary 15.2.9
Entrywise positive	$\tilde{O}(n^2/\varepsilon)$	Sinkhorn, [DGK18], ℓ^2 from [KK93; KLRS07]
	$\tilde{O}(n^2/\varepsilon^2)$	Sinkhorn, via KL [ANR17; CK21]
	$\tilde{O}(n^2)$	box-constrained, via ℓ^2 [ALOW17; CMTV17]
	$\tilde{O}(n^{1.5}/\varepsilon^2)$	Sinkhorn, quantum, Corollary 14.2.6
	$\tilde{O}(n^{1.5}/\varepsilon)$	box-constrained, quantum, Corollary 15.2.10

Table 13.1.: State-of-the-art time complexity of first- and second-order methods for finding an ε - ℓ^1 -scaling to arbitrary marginals. The boldface lines are our results, and are the only quantum algorithms for scaling that we are aware of. Here h is the smallest integer such that hr and hc are integer vectors; m is an upper bound on the number of non-zero entries of \mathbf{A} ; κ represents the ratio between the largest and the smallest entries of the optimal scalings \mathbf{X} and \mathbf{Y} , which can be exponential in n . Many referenced results originally use a different error model (e.g., ℓ^2 or Kullback-Leibler divergence), which we convert to guarantees in the ℓ^1 -norm for comparison. Here the \tilde{O} -notation hides polylogarithmic factors in n , $1/\varepsilon$ and $1/\mu$.

For matrix balancing, we say that a matrix \mathbf{A} is ε - ℓ^1 -balanced if

$$\|\mathbf{r}(\mathbf{A}) - \mathbf{c}(\mathbf{A})\|_1 \leq \varepsilon \sum_{i,j=1}^n A_{ij}, \quad (13.1.2)$$

and the goal of the (ℓ^1 -)matrix balancing problem is to find a positive diagonal matrix \mathbf{X} such that $\mathbf{B} = \mathbf{XAX}^{-1}$ is ε - ℓ^1 -balanced.

We tabulate the best-known results for matrix balancing in Table 13.2, and we expand on the methods below.

³For ℓ^1 -scaling, ℓ^1 -balancing and squared-Hellinger-balancing, the problem becomes trivial as soon as $\varepsilon \geq 2$.

⁴Their proofs work only for input matrices that are exactly scalable. However, with our potential gap bound (Theorem 14.1.1) we can generalize their analysis to work for arbitrary asymptotically-scalable matrices.

13. Matrix scaling and matrix balancing

Error ε	Time complexity	References and remarks
ℓ^1	$\tilde{O}(m \min(d, 1/\varepsilon)/\varepsilon)$	random Osborne [AP23]
	$\tilde{O}(m \log(\kappa))$	box-constrained, via ℓ^2 [CMTV17]
	$\tilde{O}(m^{1.5})$	interior-point method, via ℓ^2 [CMTV17]
	$\tilde{O}(\sqrt{mn}/\varepsilon^3)$	random Osborne, quantum, Theorem 14.4.6
ℓ^2	$\tilde{O}(m + n/\varepsilon^2)$	random Osborne [ORY17]
	$\tilde{O}(\sqrt{mn} \log(\kappa)^{1.5}/\varepsilon)$	box-constrained, quantum, Corollary 15.3.8

Table 13.2.: State-of-the-art time complexity of first- and second-order methods for finding an ε - ℓ^1 - or ℓ^2 -balancing. The boldface lines are our results, and the only quantum algorithms for scaling that we are aware of. Here κ represents the ratio between the largest and the smallest entries of the optimal scalings \mathbf{X} and \mathbf{Y} , which can be exponential in n , and d is the diameter of the directed graph with the same support as the matrix (in the classical setting, one can efficiently reduce to balancing the graph's strongly connected components, hence d is finite). Note that the results from [CMTV17] have a polylogarithmic dependence on $1/\varepsilon$, hence naturally apply to the ℓ^2 -setting as well. Here the $\tilde{O}(\cdot)$ hides polylogarithmic factors in n , $1/\varepsilon$ and $1/\mu$.

First-order methods: Sinkhorn and Osborne

Given the importance of good matrix scalings and balancings, how efficiently can we actually find them? For concreteness, let us first focus on scaling. Note that left-multiplying \mathbf{A} with a diagonal matrix \mathbf{X} corresponds to multiplying the i -th row of \mathbf{A} with X_{ii} . Hence it is very easy to get the desired row sums: just compute all row sums $r_i(\mathbf{A})$ of \mathbf{A} and define \mathbf{X} by $X_{ii} = r_i/r_i(\mathbf{A})$, then \mathbf{XA} has exactly the right row sums. Subsequently, it is easy to get the desired column sums: just right-multiply the current matrix \mathbf{XA} with diagonal matrix \mathbf{Y} where $Y_{jj} = c_j/c_j(\mathbf{XA})$, then \mathbf{XAY} will have the right column sums. The problem with this approach is, of course, that the second step is likely to undo the good work of the first step, changing the row sums away from the desired values; it is not at all obvious how to *simultaneously* get the row sums and column sums right. Nevertheless, the approach of alternating row-normalizations with column-normalizations turns out to work. This alternating algorithm is known as *Sinkhorn's algorithm* [Sin64], and has actually been (re)discovered independently in several different contexts.

It is known that the iterates in the Sinkhorn algorithm converge to an (\mathbf{r}, \mathbf{c}) -scaled matrix whenever \mathbf{A} is asymptotically (\mathbf{r}, \mathbf{c}) -scalable; for the doubly stochastic case, this was shown in [SK67]. The convergence rate of Sinkhorn's algorithm is known in various settings, and we give a brief overview of the (classical) time complexity of finding an ε - ℓ^1 -scaling, noting that a single iteration can be implemented in time $\tilde{O}(m)$. When \mathbf{A} is entrywise positive then one can scale in time $\tilde{O}(n^2/\varepsilon)$ [DGK18]; in the ℓ^2 -setting for uniform target marginals a similar result can be found in [KK93; KLR07]. In the general setting where \mathbf{A} has at most $m \leq n^2$ non-zero entries the complexity becomes $\tilde{O}(m/\varepsilon^2)$ (for arbitrary target marginals (\mathbf{r}, \mathbf{c})); a proof may be found in [ANR17] for the entrywise-positive case, in [CK21] for exactly scalable matrices (i.e., where the problem can be solved for $\varepsilon = 0$) and essentially

the same proof with a folklore potential bound (Theorem 14.1.1) yields the result for asymptotically scalable matrices.

For matrix balancing there is a similar method known as *Osborne's algorithm* [Os60; PR69]. In each iteration this chooses a row index i and defines X_{ii} such that the i -th row sum and the i -th column sum become equal. Again, because each iteration can undo the good work of earlier iterations it is not at all obvious that this converges to a balancing of \mathbf{A} . Remarkably, even though Osborne's algorithm was proposed more than six decades ago and is widely used in linear algebra software, an explicit bound on its convergence rate has only been proven recently [SS15; ORY17]! It is known to produce an ε - ℓ^1 -balancing in time $\tilde{O}(m/\varepsilon^2)$ when in each iteration the update is chosen randomly [AP23]. In [ORY17] it was shown that a weighted random variant of Osborne's algorithm produces an ε - ℓ^2 -balancing in time $\tilde{O}(m + n/\varepsilon^2)$.

Second-order methods: box-constrained Newton methods

While simple, the Sinkhorn algorithm is by no means the fastest when the parameter ε is small. The classical state-of-the-art algorithms are based on second-order methods such as (traditional) interior-point methods or so-called *box-constrained Newton methods* [CMTV17; ALOW17], the latter of which we describe in more detail below. We note that these algorithms depend on fast algorithms for graph sparsification and Laplacian system solving, so are rather complicated compared to Sinkhorn's algorithm. The box-constrained Newton methods can find ε - ℓ^1 -scaling vectors in time $\tilde{O}(mR_\infty)$, where the \tilde{O} hides polylogarithmic factors in n and $1/\varepsilon$, and R_∞ is a certain diameter bound (made precise later in the introduction). Such a result also applies to matrix balancing. For entrywise-positive matrices, R_∞ is of size $\tilde{O}(1)$, and in general it is known to be $\tilde{O}(n)$ [ALOW17, Lem. 3.3]. Alternatively, the interior-point method of [CMTV17] has a time complexity of $\tilde{O}(m^{3/2})$, which is better than the box-constrained Newton method for general inputs, but worse for entrywise-positive matrices. Our second quantum algorithm for matrix scaling (and balancing) is based on these classical box-constrained Newton methods and therefore we describe them in more detail below.

Many classical algorithms for the matrix scaling problem can be viewed from the perspective of convex optimization. For example, one can solve the matrix scaling problem by minimizing the convex (potential) function

$$f(\mathbf{x}, \mathbf{y}) = \sum_{i,j=1}^n A_{ij} e^{x_i + y_j} - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle, \quad (13.1.3)$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product on \mathbb{R}^n . Note that the partial derivative of this f w.r.t. the variable x_i is $\sum_{j=1}^n A_{ij} e^{x_i + y_j} - r_i = r_i(\mathbf{XAY}) - r_i$, and the partial derivative w.r.t. y_j is $c_j(\mathbf{XAY}) - c_j$. A minimizer \mathbf{x}, \mathbf{y} of f will have the property that all these $2n$ partial derivatives are equal to 0, which means \mathbf{XAY} is *exactly scaled*! Accordingly, (approximate) scalings can be obtained by finding (approximate) minimizers using methods from convex optimization. In fact, Sinkhorn's original algorithm can be interpreted as block coordinate descent on this f , and Osborne's algorithm can similarly be derived by slightly modifying f .

13. Matrix scaling and matrix balancing

(To see this equivalence one has to change variables and set $\mathbf{X} = \text{diag}(e^{\mathbf{x}})$ and $\mathbf{Y} = \text{diag}(e^{\mathbf{y}})$.) Sinkhorn's method is thus a first-order method.

Below we give a sketch of a second-order method, the box-constrained Newton method, that we use later to obtain an improved $\text{poly}(1/\varepsilon)$ -dependence, see Section 15.1 for details. The algorithm aims to minimize the (highly structured) convex potential function f from Eq. (13.1.3). A natural iterative method for minimizing convex functions f is to minimize in each iteration i the quadratic Taylor expansion $\frac{1}{2}\mathbf{x}^\top(\text{Hess } f(\mathbf{x}^{(i)}))\mathbf{x} + \mathbf{x}^\top\nabla f(\mathbf{x}^{(i)}) + f(\mathbf{x}^{(i)})$ of the function at the current iterate. A box-constrained method constrains the minimization of the quadratic Taylor expansion to those \mathbf{x} that lie in an ℓ^∞ -ball of radius c around the current iterate (hence the adjective "box-constrained"):

$$\mathbf{x}^{(i)} = \underset{\|\mathbf{x}-\mathbf{x}^{(i)}\|_\infty \leq c}{\text{argmin}} \frac{1}{2}\mathbf{x}^\top(\text{Hess } f(\mathbf{x}^{(i)}))\mathbf{x} + \mathbf{x}^\top\nabla f(\mathbf{x}^{(i)}).$$

This is guaranteed to decrease a convex function f whenever it is *second-order robust*, i.e., whenever the Hessian of f at a point is a good multiplicative approximation of the Hessian at every other point in a constant-radius ℓ^∞ -ball. One can show that the steps taken decrease the potential gap by a multiplicative factor which depends on the distance to the minimizer.

One then observes that the function f from Eq. (13.1.3) is second-order robust. Moreover, its Hessian has an exceptionally nice structure: it is given by

$$\text{Hess } f(\mathbf{x}, \mathbf{y}) = \left[\begin{array}{cc} \text{diag}(r(\mathbf{A}(\mathbf{x}, \mathbf{y}))) & \mathbf{A}(\mathbf{x}, \mathbf{y}) \\ \mathbf{A}(\mathbf{x}, \mathbf{y})^\top & \text{diag}(c(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \end{array} \right]$$

which is similar to a *Laplacian* matrix. This means that the key subroutine in this method (approximately) minimizes quadratic forms $\frac{1}{2}\mathbf{z}^\top\mathbf{H}\mathbf{z} + \mathbf{z}^\top\mathbf{b}$ over ℓ^∞ -balls, where \mathbf{H} is a Laplacian matrix; without the ℓ^∞ -constraint, this amounts to solving the Laplacian system $\mathbf{H}\mathbf{z} = \mathbf{b}$. Such a subroutine can be implemented for the more general class of symmetric diagonally-dominant matrices (with non-positive off-diagonal entries) on a classical computer in (almost) linear time in the number of non-zero entries of \mathbf{H} [CMTV17]. For technical reasons, one has to add a regularization term to f , and the regularized potential instead has a symmetric diagonally-dominant Hessian structure.

We give two sets of contributions in this paper: new quantum algorithms for scaling and balancing, and new quantum lower bounds showing that our algorithms are not too far from optimal. We start with the algorithms.

13.1.3. Contribution 1: quantum algorithms for matrix scaling and balancing

Because a classical scaling algorithm has to look at each non-zero matrix entry (at least with large probability), it is clear that $\Omega(m)$ is a lower bound on the classical query complexity. This would be $\Omega(n^2)$ in the case of a dense or even entrywise-positive matrix \mathbf{A} . As can be seen from Table 13.1, the best classical algorithms also achieve this m lower bound up to small factors, with various dependencies on ε . The same is true for matrix balancing (Table 13.2): $\Omega(m)$ queries are necessary, and this is achievable in different ways, with different dependencies on ε and/or other parameters.

Quantum speedups for Sinkhorn and Osborne

We first give quantum algorithms for scaling and balancing that beat the best-possible classical algorithms if $\varepsilon \in (0, 1)$ is relatively large (e.g., a small constant):

Theorem (Quantum upper bound for scaling, informal statement; see Corollary 14.1.9). *There is a quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^1 -scaling for an asymptotically-scalable $n \times n$ matrix \mathbf{A} with m non-zero entries to desired positive marginals \mathbf{r} and \mathbf{c} in time $\tilde{O}(\sqrt{mn}/\varepsilon^3)$.*

If \mathbf{A} is entrywise positive (which implies $m = n^2$), then the upper bound can be improved to $\tilde{O}(n^{1.5}/\varepsilon^2)$ (see Corollary 14.2.6).

Our scaling algorithms first achieve closeness measured in terms of the relative entropy, and then use Pinsker’s inequality ($\|\mathbf{p} - \mathbf{q}\|_1^2 = O(D(\mathbf{p}||\mathbf{q}))$) to convert this to an upper bound on the ℓ^1 -error.

Theorem (Quantum upper bound for balancing, informal statement; see Theorem 14.4.6). *There is a quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^1 -balancing for an asymptotically-balanceable $n \times n$ matrix \mathbf{A} with m non-zero entries in time $\tilde{O}(\sqrt{mn}/\varepsilon^3)$.*

Our algorithm actually achieves closeness in squared Hellinger distance, which we have converted to ℓ^1 -distance for the above statement.

Note that compared to the classical algorithms we have polynomially better dependence on n and m , at the expense of a worse dependence on ε . There have recently been a number of new quantum algorithms with a similar tradeoff: they are better than classical in terms of the main size parameter but worse in terms of the precision parameter. Examples are the quantum algorithms for solving linear and semidefinite programs [BS17; AGGW20; BKL+19; AG19] and for boosting of weak learning algorithms [AM20; RHR+21; IdW23].

Conceptually our algorithms are quite simple: we implement the Sinkhorn and Osborne algorithms but replace the exact computation of each row and column sum by *quantum amplitude estimation*. For this computation, we use the results of Chapter 12: we can approximate the sum of n numbers up to some small multiplicative error δ (with high probability) at the expense of roughly \sqrt{n}/δ queries to those numbers, and a similar number of other operations.

Our analysis is based on a potential argument (for Sinkhorn we use the above-mentioned potential f from Eq. (13.1.3)). The approximation errors δ cause us to make less progress in each iteration compared to an “exact” version of Sinkhorn or Osborne. If δ is too large then we may even make backwards progress, while if δ is very small there is no quantum speed-up! We show that there is a choice of δ for which the negative contribution due to the approximation errors is of the same order as the progress in the “exact” version, and that choice also results in a speed-up. We should caution, however, that it is quite complicated to actually implement this idea precisely and to keep track of and control the various approximation errors and failure probabilities induced by the quantum estimation algorithms, as well as by the fact that we cannot represent the numbers involved with infinite precision.⁵

⁵This issue of precision is sometimes swept under the rug in classical research on scaling algorithms.

Quantum speedups for box-constrained Newton methods

Recall from Section 13.1.2 that the box-constrained Newton method for matrix scaling heavily depended on graph sparsification and Laplacian system solving. Given the recent quantum algorithm for these problems by Apers and de Wolf [AW22], one may hope to obtain a quantum speed-up for the box-constrained Newton method. We show that one can indeed achieve this by first using the quantum algorithm for graph sparsification, and then using the classical method for the minimization procedure. We note, however, that in order to achieve a quantum speed-up in terms of m and n , we incur a polynomial dependence in the time complexity on the precision with which we can approximate \mathbf{H} and \mathbf{b} (as opposed to only a *polylogarithmic* dependence classically). Such a speed-up with respect to one parameter (dimension) at the cost of a slowdown with respect to another (precision) is more common in recent quantum algorithms for optimization problems and typically requires a more careful analysis of the impact of approximation errors. Interestingly, for the classical box-constrained Newton method, the minimization subroutine is the bottleneck, whereas in our quantum algorithm, the cost of a single iteration is dominated by the time it takes to approximate the vector \mathbf{b} . Using similar techniques as in the quantum versions of Sinkhorn, one can obtain an approximation of \mathbf{b} with ℓ^1 -error at most $\delta \|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$, in time roughly \sqrt{mn}/δ . To obtain an efficient quantum algorithm we therefore need to control $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ throughout the run of the algorithm. We do so efficiently by testing in each iteration whether the 1-norm of $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is too large; if it is, then we divide the matrix by 2 (by shifting \mathbf{x} by an appropriate multiple of the all-ones vector), which reduces the potential.

We have the following results for scaling and balancing.

Theorem (Quantum second-order upper bound for scaling, informal statement; see Corollary 15.2.9). *There is a quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^1 -scaling for an asymptotically-scalable $n \times n$ matrix \mathbf{A} with m non-zero entries to desired positive marginals \mathbf{r} and \mathbf{c} with $\|\mathbf{r}\|_1 = 1 = \|\mathbf{c}\|_1$ in time $\tilde{O}(R_\infty^{1.5} \sqrt{mn}/\varepsilon)$, where R_∞ is a diameter bound.*

Theorem (Quantum second-order upper bound for balancing, informal statement; see Corollary 15.3.8). *There is a quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^2 -balancing for an asymptotically-balanceable $n \times n$ matrix \mathbf{A} with m non-zero entries in time $\tilde{O}(R_\infty^{1.5} \sqrt{mn}/\varepsilon)$, where R_∞ is a diameter bound.*

Again, the above scaling algorithm for actually gives a guarantee with respect to relative entropy. Unfortunately, in the balancing setting, we do not get a good guarantee in terms of squared Hellinger distance or ℓ^1 -norm (as is the case for Osborne's algorithm); instead, the natural error bound one obtains is for an ℓ^∞ -version of the squared Hellinger distance, namely $\max_\ell (\sqrt{r_\ell} - \sqrt{c_\ell})^2$.

For the special case of entrywise-positive matrices \mathbf{A} , one can prove a polylogarithmic upper bound on R_∞ , which then disappears in the $\tilde{O}(\cdot)$ (see Corollaries 15.2.10 and 15.3.8). In this case one finds an ε - ℓ^2 -scaling or balancing in time $\tilde{O}(n^{1.5}/\varepsilon)$, whereas classical methods take time $\tilde{O}(n^2)$ (using the box-constrained method from [CMTV17]).

13.1.4. Contribution 2: quantum lower bounds for matrix scaling and balancing

A natural question is to what extent our quantum upper bounds for scaling and balancing can be improved further. Our second set of contributions looks at the limitations of quantum algorithms.

Quantum lower bounds for scaling and balancing in the constant- ε regime

Since the output for matrix scaling has length roughly n , there is an obvious lower bound of n even for quantum algorithms. An $\tilde{O}(n)$ -time quantum algorithm would, however, still be an improvement over our best quantum algorithms, and it would be a quadratic speed-up over the best possible classical algorithm. We dash this hope here by showing that our algorithm is essentially optimal for constant ε , even for the special case of \mathbf{A} that is exactly scalable to uniform marginals:

Theorem (Quantum lower bound for scaling with constant ε , see Corollary 16.2.3). *There exists a constant $\varepsilon > 0$ such that every quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^1 -scaling for given $n \times n$ matrix \mathbf{A} that is exactly scalable to uniform marginals and has m potentially non-zero entries, has to make $\Omega(\sqrt{mn})$ queries to \mathbf{A} .*

Our proof constructs a set of instances \mathbf{A} which hide a bit string in a permutation, it shows how approximate scalings of such an \mathbf{A} allow one to recover (most of) the bit string, and then uses the adversary method [Amb02] to lower bound the number of quantum queries to the matrix needed to find that information. In particular, we show that for a permutation $\sigma \in S_n$ and $z \in \{\pm 1\}^n$, learning a large constant fraction of the entries of z takes $\Omega(n\sqrt{n})$ queries to the entries of the “signed permutation matrix” $\mathbf{P}_{\sigma,z}$ whose $(\sigma(i), i)$ -th entry is z_i and all other entries are zero.

A similar strategy yields a quantum query lower bound for matrix balancing with constant ε :

Theorem (Quantum lower bound for balancing with constant ε , see Theorem 16.3.3). *There exists a constant $\varepsilon > 0$ such that every quantum algorithm that (with probability $\geq 2/3$) finds an ε - ℓ^1 -balancing for given $n \times n$ matrix \mathbf{A} that has m potentially non-zero entries and $\|\mathbf{A}\|_1 = 1$, has to make $\Omega(\sqrt{mn})$ queries to \mathbf{A} .*

We note that proving this lower bound for balancing is somewhat more delicate compared to the lower bound for scaling: when a block-diagonal matrix is ε - ℓ^1 -balanced, that does not imply that each of the individual block is ε - ℓ^1 -balanced; if one of the blocks has very large sum of entries and is perfectly balanced, while the other blocks are imbalanced, the full matrix will be ε - ℓ^1 -balanced without the individual blocks being balanced.

Stronger quantum lower bounds for scaling and balancing in the small- ε regime

Is it possible to get a polylogarithmic dependence on ε while still retaining a polynomial speedup in terms of n and m ? We show this is not the case by proving the following theorem:

13. Matrix scaling and matrix balancing

Theorem (Quantum lower bound for scaling with small ε , informal statement; see Theorem 17.5.2 and Corollary 17.5.3). *Every quantum matrix-scaling algorithm that (with probability $\geq \exp(-n)$) finds scaling vectors for given entrywise-positive $n \times n$ -matrix \mathbf{A} with ℓ^2 -error $1/(n^2\sqrt{\ln n})$ makes at least $\Omega(n^2)$ queries to \mathbf{A} . This even holds for uniform targets and matrices with smallest entry $\Omega(1/n^2)$. In the general setting of $m \leq n^2$ non-zero entries the lower bound becomes $\tilde{\Omega}(m)$.*

The proof of this lower bound is based on a reduction from deciding whether bit strings have Hamming weight $n/2 + 1$ or $n/2 - 1$. Specifically, given k bit strings $z^1, \dots, z^k \in \{\pm 1\}^n$ for $k = \Theta(n)$, each with Hamming weight $|z^i| = n/2 + a_i$ where $a_i \in \{\pm 1\}$, we show that any matrix scaling algorithm can be used to determine all the a_i . One can show that every quantum algorithm that computes all the a_i 's needs to make $\Omega(nk)$ quantum queries to the bit string z^1, \dots, z^k , even if the algorithm has only exponentially small (in k) success probability: to determine a single a_i with success probability at least $2/3$, one needs to make $\Omega(n)$ quantum queries to the bit string z^i [BBC+01; NW99; Amb02], and one can use the strong direct product theorem of Lee and Roland [LR13] to prove the lower bound for computing all k a_i 's simultaneously, even with only exponentially small success probability.

To convert the problem of computing the a_i to an instance of matrix scaling, one constructs a $2k \times n$ matrix \mathbf{A} whose first k rows are (roughly) given by the vectors $1 + z^i/b$ for some $b \geq 2$, and whose last k rows are given by $1 - z^i/b$. For such an \mathbf{A} , the column sums are all $2k$, and the row sums are determined by the a_i . If the matrix \mathbf{A}' obtained by a single Sinkhorn step from \mathbf{A} (i.e., rescaling all the rows) were exactly column scaled, then the *optimal* scaling factors encode the a_i . We show that, if one randomly (independently for each i) permutes the z^i beforehand, this is approximately the case: the column sums of this \mathbf{A}' will be close to the desired column sums with high probability, and hence the first step of Sinkhorn gives approximately optimal scaling factors (which encode the a_i). Then, we give a lower bound on the strong convexity parameter of the potential f , to show that *all* sufficiently precise minimizers of f also encode the a_i . In other words, from sufficiently precise scaling factors, we can recover the a_i , yielding the reduction to matrix scaling, and consequently a lower bound for the matrix scaling problem.

A similar strategy gives a quantum query lower bound for matrix balancing in the small- ε regime. We state it here informally for polynomially small ℓ^1 -error ε , whereas the detailed bound (Theorem 17.6.5) assumes a specific ℓ^2 -error (which we can easily convert to ℓ^1).

Theorem (Quantum lower bound for balancing with small ε , see Theorem 17.6.5). *Every quantum algorithm that (with probability $\geq \exp(-n)$) finds an ε - ℓ^1 -balancing for polynomially small ε for given $n \times n$ matrix \mathbf{A} that has m potentially non-zero entries and $\|\mathbf{A}\|_1 = 1$, has to make $\Omega(m)$ queries to \mathbf{A} .*

We additionally study the problem of computing an ε - ℓ^1 -approximation of the vector of row sums of an ℓ^1 -normalized $n \times n$ matrix \mathbf{A} . This is a common subroutine for matrix scaling algorithms; for instance, the gradient of the potential function f from Eq. (13.1.3) that we optimize for the upper bound can be determined from the row and column sums by subtracting the desired row and column sums, so the complexity of this subroutine directly relates to the complexity of each iteration in our algorithm. We give the following lower bound for this problem.

Theorem (Informal, see Theorem 17.7.1). *For $\varepsilon \in [1/n, 1/2]$, every quantum algorithm that computes an ε - ℓ^1 -approximation of $\mathbf{r}(\mathbf{A})$ for a given matrix $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 = 1$, takes $\Omega(n^{1.5}/\sqrt{\varepsilon})$ queries to \mathbf{A} .*

Instead of reducing to $\Theta(n)$ independent instances of the majority problem (as in the lower bound for high-precision matrix scaling and balancing sketched above), we reduce to $\Theta(n)$ independent instances of the problem of deciding whether a bit-string $z \in \{\pm 1\}^n$ has Hamming weight $1/\delta \pm 1$.

13.2. Preliminaries

13.2.1. Notation and conventions

We abbreviate $\mathbb{R}_{\geq 0} = [0, \infty)$ and write $[n] = \{1, \dots, n\}$. We use \log_2 to denote the logarithm with base 2 and \ln to denote the natural logarithm with base e . We use $\mathbb{1}_S$ to denote the indicator function of a set S (typically a probabilistic event). When necessary, we use fixed-point format where numbers are represented using b_1 leading bits, b_2 trailing bits, and one bit to denote the sign. That is, a number a written in (b_1, b_2) -fixed-point format is a number of the form $a = \pm \sum_{i=-b_2}^{b_1-1} a_i 2^i$ where $a_i \in \{0, 1\}$ for all i . For $a \in \mathbb{R}$ and $\delta > 0$, a δ -additive approximation of a is a number $\hat{a} \in [a + \delta, a - \delta]$. For $a > 0$ and $\delta > 0$, a $(1 \pm \delta)$ -multiplicative approximation of a is a number $\tilde{a} \in [(1 - \delta)a, (1 + \delta)a]$.

We write vectors \mathbf{x} and matrices \mathbf{A} in boldface, but their entries x_i and A_{ij} are written in regular face. We denote by \mathbb{R}^n the n -dimensional Euclidean space by and $\mathbb{R}^{n \times n}$ be the vector space of real $n \times n$ matrices. By convention we use $\mathbf{1}$ for the all-1 vector and $\mathbf{0}$ for the all-0 vector. For a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, let $\mathbf{A}_{\ell \bullet} \in \mathbb{R}^n$ be the vector corresponding to the ℓ -th row of \mathbf{A} and $\mathbf{A}_{\bullet \ell} \in \mathbb{R}^n$ be the vector corresponding to the ℓ -th column of \mathbf{A} . We say \mathbf{x} and \mathbf{A} are *entrywise non-negative* if all of their entries are greater or equal to 0; and we say \mathbf{x} and \mathbf{A} are *entrywise-positive* if all of their entries are strictly greater than 0. We denote by $\mathbb{R}_{\geq 0}^n$ the cone of all n -dimensional entrywise non-negative vectors and by $\mathbb{R}_{\geq 0}^{n \times n}$ the cone of all $n \times n$ entrywise non-negative matrices.

We use the standard definition of big-O notation as the set of functions satisfying a given growth bound. We write $\text{polylog}(n) = \bigcup_{i=0}^{\infty} O(\ln^i(n))$. We use the big- \bar{O} notation to hide polylogarithmic factors in the variables appearing within the parentheses.

For a bit string $z \in \{0, 1\}^n$ (or $z \in \{\pm 1\}^n$), the Hamming weight $|z|$ is defined as the number of $i \in [n]$ such that $z_i = 1$. We use x_{\max} and x_{\min} to denote the largest and smallest entry of \mathbf{x} , respectively. For a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, we define the ℓ^p -norm by viewing the matrix as a vector in \mathbb{R}^{n^2} , e.g., $\|\mathbf{A}\|_1 = \sum_{i,j=1}^n |A_{ij}|$; in particular, $\|\mathbf{A}\|_p$ should *not* be confused with the Schatten p -norm.

We apply functions to vectors entry-wise, for example we abbreviate $\sqrt{\mathbf{x}} = (\sqrt{x_i})_{i \in [n]} \in \mathbb{R}_{\geq 0}^n$ for $\mathbf{x} \in \mathbb{R}_{\geq 0}^n$ and $e^{\mathbf{x}} = (e^{x_i})_{i \in [n]} \in \mathbb{R}^n$ for $\mathbf{x} \in \mathbb{R}^n$.

We write $\mathbf{A}(\mathbf{x}, \mathbf{y})$ for the matrix whose (i, j) -th entry is $A_{ij} e^{x_i + y_j}$, where \mathbf{A} is an $n \times n$ matrix and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. We also abbreviate $\mathbf{A}(\mathbf{x}) = \mathbf{A}(\mathbf{x}, -\mathbf{x})$ in the context of matrix balancing.

13.2.2. Distance measures

We will be interested in the *relative entropy* or *Kullback-Leibler divergence* (used in the analysis of the Sinkhorn algorithm) and *Hellinger distance* (used in the analysis of the Osborne algorithm) between non-negative vectors.

To define the former, we use the function $\rho: \mathbb{R}_{\geq 0} \times \mathbb{R}_{> 0} \rightarrow [0, \infty]$ given by $\rho(\mathbf{a} \parallel \mathbf{b}) = \mathbf{b} - \mathbf{a} + \mathbf{a} \ln \frac{\mathbf{a}}{\mathbf{b}}$ (with the usual conventions, in particular $0 \ln 0 = 0$). The *relative entropy* or *Kullback-Leibler divergence* $D: \mathbb{R}_{\geq 0}^n \times \mathbb{R}_{> 0}^n \rightarrow [0, \infty]$ is then defined as

$$D(\mathbf{a} \parallel \mathbf{b}) = \sum_{i=1}^n \rho(a_i \parallel b_i).$$

When \mathbf{a} and \mathbf{b} are probability distributions, this reduces to the familiar formula $D(\mathbf{a} \parallel \mathbf{b}) = \sum_{i=1}^n a_i \ln \frac{a_i}{b_i}$, but we will also consider unnormalized \mathbf{b} . In this case we still have the following version of Pinsker's inequality:

Lemma 13.2.1 (Generalized Pinsker). *Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}_{\geq 0}^n$ and assume $\|\mathbf{a}\|_1 = 1$ and $b_\ell > 0$ for all $\ell \in [n]$. Define the function $w: (-1, \infty) \rightarrow \mathbb{R}$ by $w(\beta) = \beta - \ln(1 + \beta)$. Then*

$$D(\mathbf{a} \parallel \mathbf{b}) \geq w(\|\mathbf{a} - \mathbf{b}\|_1).$$

For $\beta \in [0, 1]$, we have the estimate $w(\beta) \geq \beta^2/4$, while for $\beta \geq 1$ we have $w(\beta) \geq (1 - \ln 2)\beta$. In particular, if $\|\mathbf{a} - \mathbf{b}\|_1 \leq 1$, then $D(\mathbf{a} \parallel \mathbf{b}) \geq \|\mathbf{a} - \mathbf{b}\|_1^2/4$.

Note that we do not require \mathbf{b} to be a probability distribution. The proof we give is heavily inspired by [KLR07], which gave a lower bound on a relative entropy in terms of an ℓ^2 -distance. We start with a lemma that verifies the properties of the function w .

Lemma 13.2.2. *Let $w: (-1, \infty) \rightarrow \mathbb{R}$ be the function defined in Lemma 13.2.1. Then for $\beta \in [0, 1]$, we have $w(\beta) \geq \beta^2/4$. Furthermore, for $\beta \geq 1$, we have $w(\beta) \geq (1 - \ln 2)\beta$.*

Proof. Set $g(\beta) = \beta^2/4$. We have $w'(\beta) = 1 - 1/(1 + \beta)$ and $g'(\beta) = 2\beta/4$. On $[0, 1]$, we have

$$(1 + \beta)w'(\beta) = 1 + \beta - 1 \geq \beta(1 + \beta)/2 = (1 + \beta)g'(\beta)$$

and $w(0) = g(0)$, so we see that $w(\beta) \geq g(\beta) = \beta^2/4$ on $[0, 1]$. For the last claim, note that $w(1) = 1 - \ln 2$ and $w'(1) = \frac{1}{2} > (1 - \ln 2)$, so by convexity of w we have $w(\beta) \geq (1 - \ln 2)\beta$ for any $\beta \in (-1, \infty)$. \square

Next we prove the main inequality that will imply the generalized Pinsker inequality.

Lemma 13.2.3. *Let $\mathbf{a}, \mathbf{d} \in \mathbb{R}^n$ be vectors such that $\|\mathbf{a}\|_1 = 1$, \mathbf{a} has positive entries, and $d_\ell > -a_\ell$ for every $\ell \in [n]$. Then*

$$\sum_{\ell=1}^n d_\ell - a_\ell \ln \left(1 + \frac{d_\ell}{a_\ell} \right) \geq \|\mathbf{d}\|_1 - \ln(1 + \|\mathbf{d}\|_1).$$

Proof. The proof consists of two parts. First, we show that for any $a, d \in \mathbb{R}$ with $a > 0$ and $d > -a$, we have

$$d - a \ln \left(1 + \frac{d}{a} \right) \geq |d| - a \ln \left(1 + \frac{|d|}{a} \right). \quad (13.2.1)$$

Clearly this holds with equality if $d \geq 0$, so assume $d < 0$. The function

$$g(\beta) = -2\beta - a \ln \left(1 - \frac{\beta}{a} \right) + a \ln \left(1 + \frac{\beta}{a} \right)$$

satisfies $g(0) = 0$ and $g'(\beta) \geq 0$ on $[0, a)$, so $g(\beta) \geq 0$ on $[0, a)$. Setting $\beta = -d$ yields

$$2d - a \ln \left(1 + \frac{d}{a} \right) + a \ln \left(1 - \frac{d}{a} \right) \geq 0$$

for $d < 0$, as desired.

To finish the proof, we use Eq. (13.2.1) and see that

$$\begin{aligned} \sum_{\ell=1}^n d_{\ell} - a_{\ell} \ln \left(1 + \frac{d_{\ell}}{a_{\ell}} \right) &\geq \sum_{\ell=1}^n |d_{\ell}| - a_{\ell} \ln \left(1 + \frac{|d_{\ell}|}{a_{\ell}} \right) \\ &= \|\mathbf{d}\|_1 - \sum_{\ell=1}^n a_{\ell} \ln \left(1 + \frac{|d_{\ell}|}{a_{\ell}} \right). \end{aligned}$$

Since the function $t \mapsto \ln(1+t)$ is concave and $\|\mathbf{a}\|_1 = 1$ and $a_{\ell} > 0$, we see that

$$\sum_{\ell=1}^n a_{\ell} \ln \left(1 + \frac{|d_{\ell}|}{a_{\ell}} \right) \leq \ln \left(1 + \sum_{\ell=1}^n a_{\ell} \cdot \frac{|d_{\ell}|}{a_{\ell}} \right) = \ln(1 + \|\mathbf{d}\|_1)$$

and the desired result follows. \square

Proof of Lemma 13.2.1. By continuity, we may assume without loss of generality that \mathbf{a} has positive entries. Recall that

$$\begin{aligned} D(\mathbf{a} \parallel \mathbf{b}) &= \sum_{\ell=1}^n b_{\ell} - a_{\ell} + a_{\ell} \ln \left(\frac{a_{\ell}}{b_{\ell}} \right) \\ &= \sum_{\ell=1}^n (b_{\ell} - a_{\ell}) - a_{\ell} \ln \left(\frac{b_{\ell} - a_{\ell}}{a_{\ell}} + 1 \right). \end{aligned}$$

Set $\mathbf{d} = \mathbf{b} - \mathbf{a}$, so that $d_{\ell} = b_{\ell} - a_{\ell} > -a_{\ell}$ for every $\ell \in [n]$. Therefore, we may apply Lemma 13.2.3 to \mathbf{a} and \mathbf{d} to get

$$D(\mathbf{a} \parallel \mathbf{b}) \geq w(\|\mathbf{b} - \mathbf{a}\|_1).$$

The claimed bounds on the function w follow from Lemma 13.2.3. \square

Next, we define the (unnormalized) *Hellinger distance* of vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}_{\geq 0}^n$ by

$$H(\mathbf{a}, \mathbf{b}) = \|\sqrt{\mathbf{a}} - \sqrt{\mathbf{b}}\|_2. \quad (13.2.2)$$

This also satisfies a lower bound in terms of the ℓ^1 -distance:

13. Matrix scaling and matrix balancing

Lemma 13.2.4 (Lower bound on Hellinger distance). *Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}_{\geq 0}^n$ with at least one of the two vectors being non-zero. Then*

$$H(\mathbf{a}, \mathbf{b})^2 \geq \frac{\|\mathbf{a} - \mathbf{b}\|_1^2}{2(\|\mathbf{a}\|_1 + \|\mathbf{b}\|_1)}.$$

Proof. Note that we have

$$\begin{aligned} \|\mathbf{a} - \mathbf{b}\|_1^2 &= \left(\sum_{\ell=1}^n |a_\ell - b_\ell| \right)^2 \\ &= \left(\sum_{\ell=1}^n \left| \sqrt{a_\ell} - \sqrt{b_\ell} \right| \cdot \left| \sqrt{a_\ell} + \sqrt{b_\ell} \right| \right)^2 \\ &\leq \left\| \sqrt{\mathbf{a}} - \sqrt{\mathbf{b}} \right\|_2^2 \cdot \left\| \sqrt{\mathbf{a}} + \sqrt{\mathbf{b}} \right\|_2^2 \end{aligned}$$

where we used the Cauchy–Schwarz inequality in the last step. The bound then follows from

$$\left\| \sqrt{\mathbf{a}} + \sqrt{\mathbf{b}} \right\|_2^2 = \sum_{\ell=1}^n (\sqrt{a_\ell} + \sqrt{b_\ell})^2 \leq 2 \sum_{\ell=1}^n (a_\ell + b_\ell) = 2(\|\mathbf{a}\|_1 + \|\mathbf{b}\|_1).$$

where the inequality follows from the arithmetic-geometric mean inequality. \square

13.2.3. Matrix scaling and balancing

Throughout we use $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{>0}^n$ for the desired row and column marginals. Unambiguously, we also use $\mathbf{r}: \mathbb{R}_{>0}^{n \times n} \rightarrow \mathbb{R}^n$ and $\mathbf{c}: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^n$ as the functions that send an $n \times n$ -matrix to its row resp. column marginal. That is, $\mathbf{r}(\mathbf{A})$ is the vector whose i -th entry equals $r_i(\mathbf{A}) = \sum_{j=1}^n A_{ij}$, and $\mathbf{c}(\mathbf{A})$ is the vector whose j -th entry is $c_j(\mathbf{A}) = \sum_{i=1}^n A_{ij}$.

We denote by $\mathbf{A}(\mathbf{x}, \mathbf{y}) = (A_{ij} e^{x_i + y_j})_{i,j \in [n]}$ the result of rescaling the rows of a matrix \mathbf{A} by $e^{\mathbf{x}}$ and the columns by $e^{\mathbf{y}}$. We say that a matrix $\mathbf{A} \in \mathbb{R}_{>0}^{n \times n}$ is *exactly scalable* to some $(\mathbf{r}, \mathbf{c}) \in \mathbb{R}_{>0}^n \times \mathbb{R}_{>0}^n$ if there exist $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ such that

$$\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{r} \quad \text{and} \quad \mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{c}.$$

We say \mathbf{A} is ε - ℓ^p -scalable to (\mathbf{r}, \mathbf{c}) for some $\varepsilon > 0$ and $p \geq 1$ if there are $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ such that

$$\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_p \leq \varepsilon \quad \text{and} \quad \|\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{c}\|_p \leq \varepsilon.$$

Finally, $\mathbf{A} \in \mathbb{R}_{>0}^{n \times n}$ is called *asymptotically scalable* to (\mathbf{r}, \mathbf{c}) if it is ε - ℓ^1 -scalable to (\mathbf{r}, \mathbf{c}) for all $\varepsilon > 0$.

In the matrix-balancing setting we require $\mathbf{y} = -\mathbf{x}$, and the marginals are compared to each other. Thus we abbreviate $\mathbf{A}(\mathbf{x}) = \mathbf{A}(\mathbf{x}, -\mathbf{x})$, and we say that an entrywise non-negative matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ is *exactly balanceable* if there exists a vector $\mathbf{x} \in \mathbb{R}^n$ such that

$$\mathbf{r}(\mathbf{A}(\mathbf{x})) = \mathbf{c}(\mathbf{A}(\mathbf{x})).$$

We say $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ is ε - ℓ^p -balanceable for some $\varepsilon > 0$ and $p \geq 1$ if there exists an $\mathbf{x} \in \mathbb{R}^n$ such that

$$\frac{\|\mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x}))\|_p}{\|\mathbf{A}(\mathbf{x})\|_1} \leq \varepsilon.$$

We say $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ is *asymptotically balanceable* if it is ε - ℓ^1 -balanceable for all $\varepsilon > 0$.

We can now formally state the computational problems associated with matrix scaling and balancing formally.

Problem 13.2.5 (ε - ℓ^p -scaling). *Given a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ and $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{> 0}^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, find $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ such that*

$$\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_p \leq \varepsilon \quad \text{and} \quad \|\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{c}\|_p \leq \varepsilon.$$

One can use the generalized Pinsker's inequality (Lemma 13.2.1) to upper bound ℓ^1 -distance by relative entropy, and it turns out that our algorithm is most naturally analyzed with the error measured by the relative entropy. which is then at the end converted into an ℓ^1 -error as a corollary. Accordingly, we also consider the following problem:

Problem 13.2.6 (ε -relative-entropy-scaling). *Given a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ and $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{> 0}^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, find $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ such that*

$$D(\mathbf{r} \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \varepsilon \quad \text{and} \quad D(\mathbf{c} \|\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \varepsilon.$$

Finally, the matrix balancing problem is defined in a similar fashion, in ℓ^p and squared Hellinger distance (see Eq. (13.2.2)).

Problem 13.2.7 (ε - ℓ^p -balancing). *Given $p \geq 1$ and an $n \times n$ matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$, find $\mathbf{x} \in \mathbb{R}^n$ such that*

$$\frac{\|\mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x}))\|_p}{\|\mathbf{A}(\mathbf{x})\|_1} \leq \varepsilon. \tag{13.2.3}$$

Problem 13.2.8 (ε - H^2 -balancing). *Given an $n \times n$ matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$, find $\mathbf{x} \in \mathbb{R}^n$ such that*

$$\frac{H^2(\mathbf{r}(\mathbf{A}(\mathbf{x})), \mathbf{c}(\mathbf{A}(\mathbf{x})))}{\|\mathbf{A}(\mathbf{x})\|_1} \leq \varepsilon. \tag{13.2.4}$$

Assumptions on the instances

We will always make the following assumptions, which are standard in the literature. For the matrix scaling problem we will assume that \mathbf{A} is asymptotically scalable to \mathbf{r}, \mathbf{c} . Similarly, in the matrix balancing problem we will always assume that it is asymptotically balanceable. The entries of \mathbf{A} (and \mathbf{r}, \mathbf{v} for the scaling problem) will be represented by rational numbers whose denominator and numerator can be represented using $\text{polylog}(n)$ bits.

Moreover, we will always assume for convenience that the matrix \mathbf{A} has at least one non-zero entry in every row and in every column. We furthermore assume that $\|\mathbf{A}\|_1 \leq 1$ and that the non-zero entries of \mathbf{A} are at least some $\mu > 0$.⁶ In

⁶If such a bound $\mu > 0$ is unknown, it can be found (with high probability) with $O(\sqrt{m})$ queries and similar time complexity using quantum minimum-finding (Theorem 11.2.6), where m is the number of possibly non-zero entries of \mathbf{A} , see Section 13.2.4. Similarly, one can enforce $\|\mathbf{A}\|_1 \leq 1$ with high probability by first estimating $\|\mathbf{A}\|_1$ up to a constant factor and then dividing every entry by this a constant times this number.

the matrix scaling problem we will further assume the target marginals \mathbf{r}, \mathbf{c} are entrywise positive, with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, while in the matrix balancing problem we assume that the diagonal entries of \mathbf{A} are zero.⁷

13.2.4. Data structures and computational model

Input model

Here we describe how our quantum algorithms can access the entries of the input matrix \mathbf{A} . For classical algorithms the access model is the same, though of course classical algorithms cannot make queries in superposition.

We assume *sparse black-box access* to the elements of \mathbf{A} via lists of the potentially non-zero entries for each row and each column, as follows. We assume in the i -th row there are s_i^r potentially non-zero entries (the algorithm doesn't know their locations nor their values in advance), and in the j -th column there are s_j^c potentially non-zero entries, where $\sum_{i=1}^n s_i^r = \sum_{j=1}^n s_j^c = m$. If our lists only contain the non-zero entries of the matrix then m would be the total number of non-zero entries in \mathbf{A} , but our current set-up is a bit more flexible, allowing these lists to also contain some 0-entries. We will for simplicity assume these positive integers $s_1^r, \dots, s_n^r, s_1^c, \dots, s_n^c$ are known to the algorithm (either given explicitly as part of the input, or via query access), though they could also be computed efficiently by binary search as explained below.

The unitaries O_1^{row} and O_1^{col} allow us to find the indices of potentially non-zero elements of rows and columns of \mathbf{A} . Specifically:

$$\begin{aligned} O_1^{\text{row}} |i\rangle |k\rangle |b\rangle &= |i\rangle |k\rangle |b + j(i, k)\rangle && \text{for } i, k, b \in [n], \\ O_1^{\text{col}} |k\rangle |j\rangle |b\rangle &= |k\rangle |j\rangle |b + i(j, k)\rangle && \text{for } j, k, b \in [n], \end{aligned}$$

where $j(i, k) \in [n]$ is the position of the k -th potentially non-zero element of row i and similarly $i(j, k) \in [n]$ is the position of the k -th potentially non-zero element of column j . The addition in the last register is modulo n (with the outcome in $[n]$ rather than in $\{0, \dots, n-1\}$). If $k > s_i^r$ then we define $j(i, k) = 0$, so if O_1^{row} maps $|i\rangle |k\rangle |b\rangle$ to itself, then we learn that $k > s_i^r$ (this is what allows us to learn s_i^r ourselves efficiently via binary search). We do the same for the columns.

We furthermore assume access to binary representations of the numerators and denominators of entries of \mathbf{A} and \mathbf{r}, \mathbf{c} in the usual way: we have unitaries $O_{\mathbf{A}}, O_{\mathbf{r}}, O_{\mathbf{c}}$ that return the numerators and denominators of the entries of $\mathbf{A}, \mathbf{r}, \mathbf{c}$. For $i, j \in [n]$, and b a string of the same number of bits as used for denominator and numerator, we have⁸

$$\begin{aligned} O_{\mathbf{A}} |i\rangle |j\rangle |b\rangle &= |i\rangle |j\rangle |b \oplus (A_{ij}^{\text{den}}, A_{ij}^{\text{num}})\rangle \\ O_{\mathbf{r}} |i\rangle |b\rangle &= |i\rangle |b \oplus (r_i^{\text{den}}, r_i^{\text{num}})\rangle \\ O_{\mathbf{c}} |j\rangle |b\rangle &= |j\rangle |b \oplus (c_j^{\text{den}}, c_j^{\text{num}})\rangle \end{aligned}$$

⁷This is without loss of generality for the following reason: If \mathbf{A} is an arbitrary matrix and \mathbf{B} denotes the matrix with the same off-diagonal entries, but zero diagonal entries, then $\mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x})) = \mathbf{r}(\mathbf{B}(\mathbf{x})) - \mathbf{c}(\mathbf{B}(\mathbf{x}))$ for any $\mathbf{x} \in \mathbb{R}^n$, while $\|\mathbf{A}(\mathbf{x})\|_1 \geq \|\mathbf{B}(\mathbf{x})\|_1$. It follows that if Eq. (13.2.3) holds for \mathbf{B} , then it also holds for \mathbf{A} .

⁸Our algorithms only require classical query access to $O_{\mathbf{r}}$ and $O_{\mathbf{c}}$, no quantum queries are needed here.

We use rational inputs so that uniform marginals can be represented exactly, as well as for the sake of consistency with the classical literature. Moreover, converting from fixed-point inputs to rational inputs is a trivial task. However, we note that inside some of our algorithms it will be useful to use fixed-point format as defined in Section 13.2.1.

Computational model

Our computational model is of a classical computer (say, a Random Access Machine for concreteness) that, in addition to its classical computations, can invoke a quantum computer. The classical computer can write to a classical-write quantum-read memory (“QCRAM”), and send a description of a quantum circuit that consists of one- and two-qubit gates from some fixed discrete universal gate set,⁹ queries to the input oracles, and queries to the QCRAM to the quantum computer. The quantum computer runs the circuit, measures the full final state in the computational basis, and returns the measurement outcome to the classical computer. We will use the QCRAM to store the scaling vectors \mathbf{x} , \mathbf{y} at any point in time in our iterative algorithms and hence will need enough QCRAM to store these $2n$ numbers up to sufficient precision (the required precision is analyzed later, in the body of the paper). The QCRAM can be queried by the quantum computer in the same way as the above unitaries O_r and O_c .

The cost of the quantum subroutines will be measured by the total number of queries to \mathbf{A} and the QCRAM, plus the number of one- and two-qubit gates. The cost of the classical computer will be measured by its number of elementary steps. This includes the cost of writing down the descriptions of the quantum circuits that the classical machine subcontracts to the quantum machine; in our algorithms these will be relatively simple, like versions of Grover’s algorithm and amplitude estimation, and hence can be written down with at most a logarithmic overhead over their number of gates. The total cost (or “time complexity”) of our algorithms is the sum of their classical and quantum costs.

13.3. Quantum subroutines for matrix scaling and balancing

In this section we build upon the approximate summation subroutine from Section 12.4 to build other subroutines for our algorithms for matrix scaling and balancing. In particular, we implement a subroutine for computing log-sum-exp quantities in Section 13.3.1 and subroutines for testing scalings and balancings in Section 13.3.2. We also recall a result on quantum graph sparsification from [AW22] in Section 13.3.3. These subroutines will be used in Chapters 14 and 15 for our algorithms for matrix scaling and balancing. Detailed implementations of the subroutines are delayed until Section 13.3.4.

⁹For concreteness assume our gate set contains the Hadamard gate, T-gate, Controlled-NOT, and 2-qubit controlled rotations over angles $2\pi/2^s$ for positive integers s (these controlled rotations are used in the circuit for the quantum Fourier transform (QFT), which we invoke later in the paper).

Subroutine LogSumExp(\mathbf{a} , r , \mathbf{y} , δ , b_1 , b_2 , η , μ)

Input: Query access to rational $\mathbf{a} \in [0, 1]^n$, rational $r \in (0, 1]$, query access to $\mathbf{y} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point format, precision $\delta \in (0, 1]$, failure probability $\eta \in (0, 1]$, lower bound $\mu > 0$ on the non-zero entries of \mathbf{a} .

Output: A number x encoded in (b_1, b_2) -fixed-point format.

Guarantee: If $b_1 \geq \lceil \log_2(|\ln(\sum_{j=1}^n a_j e^{y_j}/r)|) \rceil$ and $b_2 \geq \lceil \log_2(1/\delta) \rceil$, then with probability at least $1 - \eta$, x is a δ -additive approximation of $\ln(\sum_{j=1}^n a_j e^{y_j}/r)$.

We recall the main result from Section 12.4: given quantum query access to a vector $\mathbf{v} \in [0, 1]^n$, the goal is to compute an $\tilde{s} \geq 0$ such that $(1 - \delta)s \leq \tilde{s} \leq (1 + \delta)s$, where $s = \sum_{i=1}^n v_i$. If $\delta = \Omega(1/n)$, then this can be done using $\tilde{O}(\sqrt{n/\delta})$ quantum queries and a similar time complexity. We show that this subroutine can also be used to provide additive approximations of $\ln \sum_{i=1}^n e^{y_i}$ for $\mathbf{y} \in \mathbb{R}^n$, with only a polynomial dependence on the bit complexity of the y_i . These subroutines can also be used to test whether a matrix is approximately scaled or balanced (in a way that is essentially optimal by Theorem 17.7.1).

The ApproxSum can be easily used to implement a subroutine which computes an ℓ^1 -approximation of the vector of row (or column) marginals of a non-negative matrix, with the following guarantees.

Corollary 13.3.1. *Let $\delta, \eta > 0$, sparse oracle access to $\mathbf{A} \in [0, 1]^{n \times n}$ with m possibly non-zero entries, accessible in $(0, b)$ -fixed-point format. Then there exists an algorithm ApproxMarginals which given access to \mathbf{A} , b , δ , η outputs with probability $\geq 1 - \eta$ a vector $\tilde{\mathbf{r}} \in [0, 1]^n$ such that $\|\tilde{\mathbf{r}} - \mathbf{r}(\mathbf{A})\|_1 \leq \delta \|\mathbf{A}\|_1$. It uses $O(\sqrt{\frac{mn}{\delta}} \log(\frac{1}{\eta}))$ quantum queries and $O(\sqrt{\frac{mn}{\delta}} \log(\frac{1}{\eta}) \text{poly}(b, \log(n), \log(1/\delta)))$ other gates.*

Proof. Use ApproxSum to compute for each $i \in [n]$ an approximation \tilde{r}_i of r_i such that $|\tilde{r}_i - r_i(\mathbf{A})| \leq \delta r_i(\mathbf{A})$. Summing these estimates yields the desired inequality $\|\tilde{\mathbf{r}} - \mathbf{r}(\mathbf{A})\|_1 \leq \delta \|\mathbf{A}\|_1$. The cost of a call to ApproxSum scales with the number of possibly non-zero elements s_i in row i as $\sqrt{s_i}$; hence the total cost scales with $\sum_{i=1}^n \sqrt{s_i} \leq \sqrt{n} \sum_{i=1}^n s_i = \sqrt{mn}$. \square

13.3.1. Quantum LogSumExp

We next discuss the log-sum-exp function $\text{LSE}(y_1, \dots, y_n) = \ln(\sum_{i=1}^n e^{y_i})$, which is a basic primitive used in a wide variety of contexts, for example as a smooth approximation of the maximum function in machine learning. We will be interested in a very slight generalization, namely $\ln(\sum_{i=1}^n a_i e^{y_i}/r)$, which we shall also refer to as ‘LogSumExp’ and which arises naturally, e.g., in geometric programming; for us, it captures the row or column sums of rescaled or rebalanced matrices. The following theorem states our result for computing additive approximations to LogSumExp.

Theorem 13.3.2 (Approximate LogSumExp). *There is a quantum algorithm that implements the subroutine LogSumExp using $O(\sqrt{\frac{n}{\delta}} \log(\frac{1}{\eta}))$ queries and $\tilde{O}(\sqrt{\frac{n}{\delta}} \log(\frac{1}{\eta}))$*

Subroutine TestScaling($\mathbf{A}, \mathbf{r}, \mathbf{c}, \mathbf{x}, \mathbf{y}, \delta, b_1, b_2, \eta, \mu$)

Input: Query access to rational $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$, rational $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, query access to $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point format, precision $\delta \in (0, 1]$, desired failure probability $\eta \in [0, 1]$, lower bound $\mu > 0$ on the non-zero entries of \mathbf{A} .

Output: True or False.

Guarantee: If $b_1 \geq \log_2(|\ln(\sum_{j=1}^n A_{\ell j} e^{y_j} / r_\ell)|)$ and $b_1 \geq \log_2(|\ln(\sum_{i=1}^n A_{i\ell} e^{x_i} / c_\ell)|)$ for all $\ell \in [n]$, and if also $b_2 \geq \lceil \log_2(1/\delta) \rceil$, then with failure probability $\leq \eta$ the output is **True** if both $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ and $D(\mathbf{c} \parallel \mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ are $\leq \delta$, and it is **False** if either is $\geq 2\delta$.

other gates. The \tilde{O} -notation hides polynomial factors in b_1, b_2 , and the encoding length of \mathbf{a} , as well as polylogarithmic factors in $n, 1/\delta, 1/\mu, 1/r$.

It is clear that a multiplicative approximation of $\sum_{i=1}^n v_i$ where $v_i = a_i e^{y_i} / r$ yields an additive approximation of $\ln(\sum_{i=1}^n a_i e^{y_i} / r)$. However, one obstacle is that one cannot simply compute all exponentials to sufficient precision and use exact or approximate summation on the result. Indeed, in general one would even need space exponential in the bitsize of y_i to represent e^{y_i} to constant precision. Instead, we compute a multiplicative estimate of the sum $\sum_{j=1}^n w_j$ where $w_j = \frac{a_j}{a_{j_*}} e^{y_j - y_{j_*}}$ for some $j_* \in [n]$. This approach is widely used in practice in classical computing (see e.g. [AP23]). For our implementation of LogSumExp, we choose j_* as an index $j \in [n]$ for which $a_j e^{y_j}$ is maximal, which we find using quantum max-finding (Theorem 11.2.6). Note that comparisons between such numbers can also be implemented efficiently. This ensures that all the relative quantities w_j are in $[0, 1]$. Then we can use ApproxSum (Corollary 12.4.4) to compute an estimate of their sum, and use the identity $\ln(\sum_{j=1}^n w_j) = \ln(\sum_{j=1}^n a_j e^{y_j} / r) + \ln(r) - \ln(a_{j_*}) - y_{j_*}$ to compute the desired log-sum-exp. We implement everything such that the fixed-point format (b_1, b_2) for both the input and output of the queries is the same, avoiding the need to change the encoding format in every iteration of the algorithm. We prove Theorem 13.3.2 in Line 3, paying special attention to the bit-complexity required by each operation.

13.3.2. Quantum subroutines for testing scalings and balancings

As a consequence of Theorem 13.3.2, we can also test whether a rescaled matrix has desired marginals up to some precision as measured in relative entropy, as described in the subroutine TestScaling. We prove Theorem 13.3.3 in Line 11.

Theorem 13.3.3 (TestScaling). *The subroutine TestScaling can be implemented using one call to a subroutine for obtaining a multiplicative estimate of the sum of all matrix entries and $2n$ calls to LogSumExp. Accordingly, there is a quantum algorithm that implements TestScaling using $\tilde{O}(\sqrt{mn}/\delta \log(1/\eta))$ queries and other gates, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1 and b_2 , and polylogarithmic factors in n, m , and δ .*

A similar result for ℓ^1 -scaling can be directly obtained from Corollary 13.3.1. For matrix balancing, we can implement a similar subroutine TestBalancing

13. Matrix scaling and matrix balancing

which tests whether a matrix is approximately balanced in squared Hellinger distance, with guarantees as follows. The proof is given in Line 11. Note that similarly as in `TestScaling`, testing can only be done approximately, and should be interpreted as outputting `True` when the squared Hellinger distance is $\leq \delta$, and outputting `False` when the squared Hellinger distance is $\geq 2\delta$.

Proposition 13.3.4 (`TestBalancing`). *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with zeroes on the diagonal, each row and column containing at least one non-zero element and all non-zero entries at least $\mu > 0$. Let $\delta, \eta \in (0, 1)$ be rational numbers. Then there exists an algorithm `TestBalancing` that when given query access to $\mathbf{x} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point-format and η, δ as input, determines with success probability $\geq 1 - \eta$ whether $\mathbf{A}(\mathbf{x})$ is δ - H^2 -balanced, and uses $\tilde{O}(\sqrt{mn}/\delta \log(1/\eta))$ queries and other gates, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1 and b_2 , and polylogarithmic factors in n, m , and δ .*

13.3.3. Quantum graph sparsification

The last ingredient we require is a result about quantum graph sparsification from [AW22]. Consider an undirected graph $G = (V, E, w)$ with vertex set $V = [n]$ and nonnegative edge-weights given by $w: E \rightarrow \mathbb{R}_{\geq 0}$. The *Laplacian* L_G of G is the $n \times n$ matrix $L_G = \sum_{(u,v) \in E} w(u,v)(e_u - e_v)(e_u - e_v)^T$. This matrix has the weighted degrees $d_u = \sum_{v:(u,v) \in E} w(u,v)$ of the graph on its diagonal entries, and the negated weights $-w(u,v)$ on its off-diagonal entries. A graph $H = (V, E', w')$ with $E' \subseteq E$ is called an ε -spectral sparsifier of G , if the Laplacians of G and H are close in the following sense: for every $x \in \mathbb{R}^n$ it holds that

$$(1 - \varepsilon)x^T L_G x \leq x^T L_H x \leq (1 + \varepsilon)x^T L_G x. \quad (13.3.1)$$

Alternatively, we can rewrite this as $(1 - \varepsilon)L_G \leq L_H \leq (1 + \varepsilon)L_G$, where $A \leq B$ denotes that $B - A$ is positive semidefinite.

Theorem 13.3.5 (Quantum algorithm for graph sparsification [AW22]). *For every $\varepsilon > 0$ there exists a quantum algorithm that, given adjacency-list access to a weighted and undirected graph G with n vertices and m edges, outputs (with success probability $\geq 2/3$) an ε -spectral sparsifier H of G with $\tilde{O}(n/\varepsilon^2)$ edges, using $\tilde{O}(\sqrt{mn}/\varepsilon)$ queries and other elementary operations. The algorithm uses $O(\log n)$ qubits and a QRAM of $\tilde{O}(\sqrt{mn}/\varepsilon)$ bits.*

Note that the algorithm outputs a classical description of the sparse graph H , for instance as a list of the remaining $\tilde{O}(n/\varepsilon^2)$ edges and their new weights. [AW22] also show that their algorithm is optimal in terms of m, n , and $\varepsilon > \sqrt{n/m}$, up to polylogarithmic factors. In contrast, The classical complexity of finding an ε -spectral sparsifier is $\tilde{O}(m \log(1/\varepsilon))$.

13.3.4. Detailed implementations

In this subsection we give detailed proofs of the results on approximate summing described in this section. We will take for granted the fact that there exist arithmetic circuits for computing ratios, exponentials, logarithms, and trigonometric functions,

with the following property: for all inputs encoded in (b_1, b_2) -fixed-point format for which the number to be computed can be encoded in (b_3, b_4) -fixed-point format with additive error at most 2^{-b_4} , the output of the circuit is such an additive approximation. Furthermore, these circuits have size at most polynomial in b_1, b_2, b_3 , and b_4 . For exponentials and logarithms, one may achieve this (for instance) by using repeated squaring and Taylor series approximations; see e.g. [BZ11]. This assumption also implies that given a rational number, encoded by its numerator and denominator, we can efficiently compute a fixed-point representation of this rational number to any desired additive precision (by treating the numerator and denominator as fixed-point numbers and computing their ratio), as long as the number of leading bits for the output format is chosen to be sufficiently large.

We first describe a classical algorithm, `RelativeEntryAdditiveApprox` in Algorithm 13.1, that efficiently computes the ratio of two numbers of the form ae^y up to a certain precision.

Algorithm 13.1: `RelativeEntryAdditiveApprox`($a_1, a_2, y_1, y_2, b, b_1, b_2, c, d$)

Input: Non-negative numbers $a_1, a_2 \in [0, 1 - 2^{-b}]$ encoded in $(0, b)$ fixed-point format, numbers $y_1, y_2 \in \mathbb{R}$ encoded in (b_1, b_2) fixed-point format, natural numbers c, d

Output: A non-negative number in (d, c) -fixed-point format.

Analysis: Lemma 13.3.6

```

1 if  $a_2 = 0$  then
2   | return  $2^d - 2^{-c}$ ;
3 end if
4 if  $a_1 = 0$  then
5   | return 0;
6 end if
7 compute  $\Delta \leftarrow y_1 - y_2$ ;
8 if  $\Delta > b + d$  then
9   | return  $2^d - 2^{-c}$ ;
10 else if  $\Delta < -b - c$  then
11   | return 0;
12 else
13   | compute estimate  $\alpha \geq 0$  of  $e^\Delta$  encoded in  $(2(b + d), b + c + 3)$  fixed-point
      | format;
14   | compute estimate  $\beta \geq 0$  of  $a_1/a_2$  encoded in  $(b, 2(b + d) + c + 3)$ 
      | fixed-point format;
15   | compute  $\gamma' = \alpha \cdot \beta$  exactly encoded in  $(3b + 2d, 3b + 2d + 2c + 6)$ 
      | fixed-point format;
16   | let  $\gamma$  be the result of rounding  $\min\{\gamma', 2^d - 2^{-c}\}$  to the nearest integer
      | multiple of  $2^{-c}$ ;
17   | return  $\gamma$ ;
18 end if

```

Lemma 13.3.6. *Let $y_1, y_2 \in \mathbb{R}$ be two numbers encoded in (b_1, b_2) -fixed-point format, and let $a_1, a_2 \in [0, 1 - 2^{-b}]$ be non-negative numbers encoded in $(0, b)$ -fixed-point format.*

13. Matrix scaling and matrix balancing

Furthermore, let $c, d \geq 1$ be natural numbers. Then Algorithm 13.1 with these inputs returns a non-negative number $\gamma \in [0, 2^d - 2^{-c}]$ encoded in (d, c) -fixed-point format, such that

$$|\gamma - \min\{\frac{a_1}{a_2}e^{y_1-y_2}, 2^d - 2^{-c}\}| \leq 2^{-c}.$$

Here we use the convention that $a_1/a_2 = \infty$ whenever $a_1 \neq 0 = a_2$. The algorithm terminates in time polynomial in b, b_1, b_2, c, d .

Proof. We use the same notation as in the algorithm. The cases where $a_2 = 0$ or $a_1 = 0$ are clear. Otherwise, if $\Delta = y_1 - y_2 > b + d$ then

$$\frac{a_1}{a_2}e^\Delta > \frac{a_1}{a_2}e^{b+d} \geq 2^{-b}e^{b+d} > 2^d$$

so $\min\{e^\Delta a_1/a_2, 2^d - 2^{-c}\} = 2^d - 2^{-c}$, and returning this value is a correct output. Similarly, if $\Delta < -b - c$, then

$$\frac{a_1}{a_2}e^\Delta < \frac{a_1}{a_2}e^{-b-c} < \frac{a_1}{a_2}2^{-b-c} \leq 2^{-c}$$

so 0 is a 2^{-c} -additive approximation of $e^\Delta a_1/a_2$. For the final and most interesting case, note that $e^\Delta < 2^{2(b+d)}$ and $\frac{a_1}{a_2} \leq 2^b$, so it suffices to use $2(b+d)$ and b leading bits to ensure that we include the first possibly non-trivial binary digit of e^Δ and a_1/a_2 , respectively. Due to our choice of the number of trailing bits, α and β satisfy

$$|\alpha - e^\Delta| \leq 2^{-b-c-3}, \quad |\beta - \frac{a_1}{a_2}| \leq 2^{-2(b+d)-c-3},$$

and as a consequence we see that

$$\begin{aligned} \gamma' = \alpha \cdot \beta &\leq \frac{a_1}{a_2}e^\Delta + 2^{-b-c-3}\frac{a_1}{a_2} + 2^{-2(b+d)-c-3}e^\Delta + 2^{-(3b+2d+2c+6)} \\ &\leq \frac{a_1}{a_2}e^\Delta + 2^{-c-3} + 2^{-c-3} + 2^{-c-3} \\ &\leq \frac{a_1}{a_2}e^\Delta + 2^{-c-1} \end{aligned}$$

where we used $\frac{a_1}{a_2} \leq 2^b$ and $e^\Delta \leq 2^{2(b+d)}$ in the second inequality. Similarly, we obtain the lower bound

$$\begin{aligned} \gamma' = \alpha \cdot \beta &\geq \frac{a_1}{a_2}e^\Delta - 2^{-b-c-3}\frac{a_1}{a_2} - 2^{-2(b+d)-c-3}e^\Delta - 2^{-(3b+2d+2c+6)} \\ &\geq \frac{a_1}{a_2}e^\Delta - 2^{-c-3} - 2^{-c-3} - 2^{-4b-2c-6} \\ &\geq \frac{a_1}{a_2}e^\Delta - 2^{-c-1} \end{aligned}$$

again using $\frac{a_1}{a_2} \leq 2^b$ and $e^\Delta \leq 2^{2(b+d)}$ in the second inequality. The quantity $\gamma' = \alpha \cdot \beta$ can be computed exactly using $3(b+d)$ leading bits and $3b+2d+2c+6$ trailing bits, and is guaranteed to be a 2^{-c-1} -additive approximation of $e^\Delta a_1/a_2$. Therefore, $\min\{\gamma', 2^d - 2^{-c}\}$ is a 2^{-c-1} -additive approximation of $\min\{e^\Delta a_1/a_2, 2^d - 2^{-c}\}$, and rounding to the nearest integer multiple of 2^{-c} incurs an additional additive error of at most 2^{-c-1} , so $|\gamma - \min\{e^\Delta a_1/a_2, 2^d - 2^{-c}\}| \leq 2^{-c}$. \square

As a corollary we obtain another algorithm, `GreaterOrEqual` in Algorithm 13.2, which allows us to compare two numbers of the form ae^y . Note that the comparisons cannot be exact, since we cannot compute these numbers explicitly. We therefore allow our algorithm to return an “incorrect” result if two numbers of this form are (multiplicatively) close, which suffices for the purposes of our later algorithms.

Algorithm 13.2: `GreaterOrEqual`($a_1, a_2, y_1, y_2, b, b_1, b_2, c$)

Input: Access to non-negative numbers $a_1, a_2 \in [0, 1 - 2^{-b}]$ with entries encoded in $(0, b)$ fixed-point format, numbers $y_1, y_2 \in \mathbb{R}$ encoded in (b_1, b_2) fixed-point format, a natural number $c \geq 1$

Output: A boolean.

Analysis: Corollary 13.3.7

- 1 **if** $(a_1, y_1) = (a_2, y_2)$ **then return True**;
 - 2 $\gamma \leftarrow$ `RelativeEntryAdditiveApprox` ($a_1, a_2, y_1, y_2, b, b_1, b_2, c, 1$) encoded in $(1, c)$ fixed-point format;
 - 3 **return True** if $\gamma \geq 1$; and **False** if $\gamma < 1$;
-

Corollary 13.3.7. *Algorithm 13.2 returns **True** whenever $e^{y_1 - y_2} a_1 / a_2 \geq 1 + 2^{-c}$ or $(a_1, y_1) = (a_2, y_2)$, and **False** if $e^{y_1 - y_2} a_1 / a_2 \leq 1 - 2^{-c}$, and runs in time polynomial in b, b_1, b_2, c . Here we use the convention that $a_1 / a_2 = \infty$ whenever $a_1 \neq 0 = a_2$.*

Proof of Theorem 13.3.2

In this section we analyze a quantum implementation of `LogSumExp`, and prove Theorem 13.3.2. The following lemma about converting rational numbers to fixed-point format will be useful.

Lemma 13.3.8. *Let $\mathbf{a} \in [0, 1]^n$ with smallest non-zero entry at least $\mu > 0$, and let $r \in \mathbb{R}$ be such that $\mu < r \leq 1$. For $\delta > 0$, let $b = \lceil \log_2(1/(\delta\mu)) + 2 \rceil$ and let \hat{a}_j be a $(0, b)$ -fixed-point encoding of a_j . Then for every $\mathbf{y} \in \mathbb{R}^n$ we have*

$$\left| \ln \left(\sum_{j=1}^n \hat{a}_j e^{y_j} \right) - \ln \left(\sum_{j=1}^n a_j e^{y_j} \right) \right| \leq \frac{\delta}{4}.$$

Proof. We show that the converting \mathbf{a} from rational to fixed-point format by rounding up to the nearest integer multiple of 2^{-b} does not change the logarithm of the sum of $a_j e^{y_j}$ by much. For every $j \in [n]$, the fixed-point encoding \hat{a}_j of a_j is guaranteed to be a $\frac{\delta\mu}{8}$ -additive approximation of a_j , which is zero if and only if $a_j = 0$. Since the non-zero entries of \mathbf{a} are lower bounded by μ , we know that \hat{a}_j is a $(1 \pm \frac{\delta}{8})$ -multiplicative approximation of a_j for every $j \in [n]$. In particular, this shows that $\sum_{j=1}^n \hat{a}_j e^{y_j}$ is a $(1 \pm \frac{\delta}{8})$ -multiplicative approximation of $\sum_{j=1}^n a_j e^{y_j}$. This multiplicative error implies the following additive error

$$\left| \ln \left(\sum_{j=1}^n \hat{a}_j e^{y_j} \right) - \ln \left(\sum_{j=1}^n a_j e^{y_j} \right) \right| \leq \ln \left(1 + \frac{\delta}{8} \right) \leq \frac{\delta}{4}$$

where the last inequality uses the bound $|\ln(1 + z)| \leq 2|z|$ for $z \in [-1/2, 1/2]$. \square

13. Matrix scaling and matrix balancing

Next we discuss our quantum implementation of `LogSumExp`, which is given in Algorithm 13.3, and prove Theorem 13.3.2, which we restate for convenience.

Algorithm 13.3: Quantum implementation of `LogSumExp`($\mathbf{a}, r, \mathbf{y}, \delta, b_1, b_2, \eta, \mu$)

Input: Query access to rational $\mathbf{a} \in [0, 1]^n$, rational $r \in (0, 1]$, query access to $\mathbf{y} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point format, precision $\delta \in (0, 1]$, failure probability $\eta \in (0, 1]$, lower bound $\mu > 0$ on the non-zero entries of \mathbf{a} .

Output: A number x encoded in (b_1, b_2) -fixed-point format.

Analysis: Theorem 13.3.2

- 1 set $b = \lceil \log_2(1/(\delta\mu)) + 2 \rceil$ and replace the query access to \mathbf{a} by a unitary which maps j to a_j , encoded in $(0, b)$ fixed-point format;
 - 2 replace r by the encoding of r in $(0, \lceil \log_2(1/(r\delta)) + 2 \rceil)$ fixed-point format;
 - 3 set $\delta' = \delta/2$;
 - 4 set $c = \lceil \log_2(n/\delta') \rceil + 6$;
 - 5 find with quantum max-finding (Theorem 11.2.6) a j^* such that $e^{y_j - y_{j^*}} a_j / a_{j^*} \leq 3/2$ for all $j \in [n]$, using `GreaterOrEqual`($a_i, a_j, y_i, y_j, b, b_1, b_2, 1$) for comparison, with failure probability $\eta/2$;
 - 6 let U_v give query access to the vector v with entries $v_j = \frac{1}{2} \text{RelativeEntryAdditiveApprox}(a_j, a_{j^*}, y_j, y_{j^*}, b, b_1, b_2, c, 1)$;
 - 7 use `ApproxSum` (Corollary 12.4.4) with U_v to compute $S' = \sum_{j=1}^n v_j$ in $(\lceil \log_2(n) \rceil + 1, c + 1)$ fixed-point format, with $\lambda = 6$, failure probability $\eta/2$ and multiplicative error $\delta'/32$;
 - 8 compute estimate $\alpha \geq 0$ of $\ln(2S')$ in $(\lceil \log_2(\log_2(n) + 2) \rceil, \lceil \log_2(1/\delta') + 3 \rceil)$ fixed-point format;
 - 9 compute estimate β of $\ln(r)$ in $(2\lceil \log_2 r \rceil, \lceil \log_2(1/\delta') + 3 \rceil)$ fixed-point format;
 - 10 compute estimate γ of $\ln(a_{j^*})$ in $(\lceil \log_2(\ln(1/\mu) + 1) \rceil, \lceil \log_2(1/\delta') + 3 \rceil)$ fixed-point format;
 - 11 **return** $y_{j^*} + \gamma + \alpha - \beta$ in (b_1, b_2) fixed-point format;
-

Theorem 13.3.2 (Approximate `LogSumExp`). *There is a quantum algorithm that implements the subroutine `LogSumExp` using $O(\sqrt{\frac{n}{\delta}} \log(\frac{1}{\eta}))$ queries and $\tilde{O}(\sqrt{\frac{n}{\delta}} \log(\frac{1}{\eta}))$ other gates. The \tilde{O} -notation hides polynomial factors in b_1, b_2 , and the encoding length of \mathbf{a} , as well as polylogarithmic factors in $n, 1/\delta, 1/\mu, 1/r$.*

Proof. We analyze Algorithm 13.3. The details for rounding the input vector \mathbf{a} to fixed-point format are dealt with in Lemma 13.3.8, from now on we assume that each a_j is encoded in $(0, b)$ -fixed-point format. Assume an index j^* as stated in Line 5 of the algorithm is indeed found. For each $j \in [n]$, let ξ_j be what would be the classical result of the j -th call to `RelativeEntryAdditiveApprox` on Line 6 in Algorithm 13.3. Note that in the algorithm, such calls are not made individually but in superposition by `ApproxSum`; however, the ξ_j are well-defined, as `RelativeEntryAdditiveApprox` is a deterministic subroutine. Then for all $j \in [n]$, ξ_j satisfies

$$|\xi_j - e^{y_j - y_{j^*}} a_j / a_{j^*}| \leq \frac{\delta'}{64n}, \quad (13.3.2)$$

as $c \geq \log_2(n/\delta') + 6$ and $e^{y_j - y_{j^*}} a_j / a_{j^*} \leq 2^1 - 2^{-c}$. For convenience, we write $\chi_j = e^{y_j - y_{j^*}} a_j / a_{j^*}$. The number S' returned by quantum approximate summing satisfies

$$S' \in \left[1 - \frac{\delta'}{32}, 1 + \frac{\delta'}{32} \right] \cdot \sum_{j=1}^n \frac{1}{2} \xi_j,$$

so in particular, $2S'$ satisfies

$$2S' \in \left[\left(1 - \frac{\delta'}{32} \right) \sum_{j=1}^n \left(\chi_j - \frac{\delta'}{64n} \right), \left(1 + \frac{\delta'}{32} \right) \sum_{j=1}^n \left(\chi_j + \frac{\delta'}{64n} \right) \right]$$

by Eq. (13.3.2). Note that $\sum_{j=1}^n (\chi_j - \delta'/(64n))$ is non-negative, since every χ_j is non-negative, $\chi_{j^*} \geq 1$ and $\delta' \leq 1$. This implies that

$$\begin{aligned} \ln(2S') &\geq \ln \left(1 - \frac{\delta'}{32} \right) + \ln \left(\sum_{j=1}^n \left(\chi_j - \frac{\delta'}{64n} \right) \right) \\ &\geq -\frac{\delta'}{16} + \ln \left(\sum_{j=1}^n \chi_j \right) + \ln \left(1 - \frac{\delta'}{64 \sum_{j=1}^n \chi_j} \right) \\ &\geq -\frac{\delta'}{16} + \ln \left(\sum_{j=1}^n \chi_j \right) + \ln \left(1 - \frac{\delta'}{64} \right) \\ &\geq -\frac{\delta'}{16} + \ln \left(\sum_{j=1}^n \chi_j \right) - \frac{\delta'}{32} \end{aligned}$$

where we have used $\sum_{j=1}^n \chi_j \geq \chi_{j^*} = 1$ in the second inequality, and $\ln(1-z) \geq -2z$ for $z \in [0, 1/2]$ in the first and third inequality. A similar computation shows that

$$\ln(2S') \leq \ln \left(\sum_{j=1}^n \chi_j \right) + \frac{\delta'}{16} + \frac{\delta'}{32}.$$

To summarize, this shows that

$$\left| \ln(2S') - \ln \left(\sum_{j=1}^n \chi_j \right) \right| \leq \frac{\delta'}{16} + \frac{\delta'}{32}.$$

Next, since $\alpha \geq 0$ is an estimate of $\ln(2S')$ with $\lceil \log_2(1/\delta') + 3 \rceil$ bits of precision, we get

$$\left| \alpha - \ln \left(\sum_{j=1}^n \chi_j \right) \right| \leq |\alpha - \ln(2S')| + \frac{\delta'}{8} \leq \frac{\delta'}{4}.$$

As we also have

$$|\beta - \ln(r)| \leq \frac{\delta'}{8}, \quad |\gamma - \ln(a_{j^*})| \leq \frac{\delta'}{8},$$

we get

$$\left| \beta - (y_{j^*} + \gamma + \alpha) - \ln \left(r / \sum_{j=1}^n a_j e^{y_j} \right) \right| \leq \frac{\delta'}{2}.$$

13. Matrix scaling and matrix balancing

Truncating the quantity $\beta - (y_{j^*} + \gamma + \alpha)$ to $b_2 \geq \lceil \log_2(1/\delta') \rceil$ bits introduces an additional error of at most $\frac{\delta'}{2}$, so the returned result is a δ' -additive approximation of $\ln(\sum_{j=1}^n a_j e^{y_j} / r)$.

For the time complexity statement, note that the expensive operations are finding the maximum of the ξ_j and approximating the sum of the ξ_j . The maximum-finding subroutine returns a correct j^* with failure probability at most η in time $\tilde{O}(\sqrt{n} \log(\frac{1}{\eta}))$, see Theorem 11.2.6. Approximating the sum can be done in time $\tilde{O}(\sqrt{\frac{n}{\delta}} \log(\frac{1}{\eta}))$ by Corollary 12.4.4. The other (arithmetic) operations can be implemented in time polynomial in b_1, b_2, b , the encoding length of α , and polylogarithmic in $n, 1/\eta, 1/\mu, 1/\delta$ and $1/r$, yielding the desired time complexity. \square

Proof of Theorem 13.3.3

In this subsection, we describe how to implement the subroutine `TestScaling`, see Algorithm 13.4.

Algorithm 13.4: Implementation of `TestScaling`($\mathbf{A}, \mathbf{r}, \mathbf{c}, \mathbf{x}, \mathbf{y}, \delta, b_1, b_2, \eta, \mu$)

Input: Query access to rational $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$, rational $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, query access to $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point format, precision $\delta \in (0, 1]$, desired failure probability $\eta \in [0, 1]$, lower bound $\mu > 0$ on the non-zero entries of \mathbf{A} .

Output: `True` or `False`.

Analysis: Theorem 13.3.3

```

1 Compute
   $\gamma \in [(1 - \delta/80) \min\{\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1, 20\}, (1 + \delta/80) \min\{\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1, 20\}]$  with
  success probability  $\geq 1 - \eta/2$ ; // See Lemma 13.3.9.
2 if  $\gamma \geq 10$  then
3   | return False;
4 end if
5 for  $\ell \in [n]$  do
6   | Compute  $a_\ell = -x_\ell - \text{LogSumExp}(\mathbf{A}_{\ell \bullet}, r_\ell, \mathbf{y}, \delta/4, b_1, b_2 + 2, \eta/4n, \mu)$ ;
  // on success,  $a_\ell$  is a  $\frac{\delta}{4}$ -additive approx. of  $\ln(r_\ell/r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ 
7   | Compute  $b_\ell = -y_\ell - \text{LogSumExp}(\mathbf{A}_{\bullet \ell}, c_\ell, \mathbf{x}, \delta/4, b_1, b_2 + 2, \eta/4n, \mu)$ ;
  // on success,  $b_\ell$  is a  $\frac{\delta}{4}$ -additive approx. of  $\ln(c_\ell/c_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ 
8 end for
9 return True if  $\gamma - 1 + \sum_{\ell=1}^n r_\ell a_\ell \leq 3\delta/2$  and  $\gamma - 1 + \sum_{\ell=1}^n c_\ell b_\ell \leq 3\delta/2$ ,
  otherwise False;
```

The first step of `TestScaling` (Line 1) checks whether $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ is at most a constant (here chosen to be 20); we discuss its quantum implementation in Lemma 13.3.9. (A classical implementation would require time $\tilde{O}(m)$.) In the remainder of the algorithm we use our quantum implementation of `LogSumExp`.

Lemma 13.3.9. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and m non-zero rational entries, each at least $\mu > 0$. Assume one is given quantum query access to \mathbf{A} and (b_1, b_2) -fixed-point*

representations of $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. Then there exists a quantum algorithm which computes a multiplicative $(1 \pm \delta)$ -approximation of $\min\{\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1, 20\}$, with success probability $\geq 2/3$, using $\tilde{O}(\sqrt{\frac{m}{\delta}})$ queries and a similar number of other operations, where the $\tilde{O}(\cdot)$ hides polylogarithmic factors in n , $1/\delta$, and polynomial factors in b_1, b_2 and the encoding length of the entries of \mathbf{A} .

Proof. We first compute the location of the largest entry $v > 0$ of the matrix $\mathbf{A}(\mathbf{x}, \mathbf{y})$ using the subroutine `GreaterOrEqual`. Quantumly, this can be done in time $\tilde{O}(\sqrt{m})$ with quantum maximum-finding Theorem 11.2.6, using `RelativeEntryAdditiveApprox` for comparisons.

We then proceed as in the implementation of `LogSumExp`; we use `RelativeEntryAdditiveApprox` and approximate summing. This gives us an $O(\delta)$ -additive approximation of $\ln(\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1/v) + \ln(v) = \ln(\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1)$. We can use this to determine whether $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ is at most 20 or at least 15. In the latter case we can use the $O(\delta)$ -additive approximation of $\ln(\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1)$ to give a multiplicative $(1 \pm \delta)$ -approximation of $\min\{\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1, 20\}$. In the former case, we can efficiently exponentiate an $O(\delta)$ -additive approximation of $\ln(\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1)$ to obtain a multiplicative approximation of $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ since we have an upper bound on its value.

Note that the above has avoided computing the largest entry v in $\mathbf{A}(\mathbf{x}, \mathbf{y})$ explicitly. This is important since v may be exponentially large in n . We can, however, compute $\ln(v)$ efficiently since it is the logarithm of an entry of \mathbf{A} plus the corresponding coordinates of \mathbf{x} and \mathbf{y} .

Finally, note that (by Lemma 13.3.6 and Corollary 12.4.4) the multiplicative $(1 \pm O(\delta))$ -approximation of $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1/v$ can be computed in time $\tilde{O}(\sqrt{m/\delta})$ quantumly. \square

We now analyze Algorithm 13.4 to prove Theorem 13.3.3, restated here for convenience.

Theorem 13.3.3 (TestScaling). *The subroutine `TestScaling` can be implemented using one call to a subroutine for obtaining a multiplicative estimate of the sum of all matrix entries and $2n$ calls to `LogSumExp`. Accordingly, there is a quantum algorithm that implements `TestScaling` using $\tilde{O}(\sqrt{mn/\delta} \log(1/\eta))$ queries and other gates, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1 and b_2 , and polylogarithmic factors in n, m , and δ .*

Proof. First observe that the choices of b_1 and b_2 are assumed to be such that the assumptions for every call to `LogSumExp` are satisfied. We use Lemma 13.3.9 to implement Line 1 with success probability $\geq 1 - \eta/2$ so that γ is a $(1 \pm \delta/80)$ -multiplicative approximation of $\min\{\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1, 20\}$. Next, note that each call to `LogSumExp` succeeds with probability at least $1 - \frac{\eta}{4n}$, so the probability of everything succeeding is at least $1 - \eta$ by a union bound. Recall that

$$\begin{aligned} D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) &= \sum_{i=1}^n \rho(r_i \parallel r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))) = \sum_{i=1}^n (r_i(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_i + r_i \ln(\frac{r_i}{r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))})) \\ &= \|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 - 1 + \sum_{i=1}^n r_i \ln\left(\frac{r_i}{r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))}\right). \end{aligned}$$

13. Matrix scaling and matrix balancing

Therefore, we may estimate $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ to additive precision $\delta/2$ by estimating $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ and $\sum_{i=1}^n r_i \ln(\frac{r_i}{r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))})$ to additive precision $\delta/4$. Since $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq 20$, obtaining a $(1 \pm \delta/80)$ -multiplicative approximation of $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ suffices to estimate it up to an additive error $\delta/4$. We can now distinguish two cases:

- (i) If $\gamma > 10 \geq 5/(1 - \delta/80)$, then we can conclude that (\mathbf{x}, \mathbf{y}) does not form a δ -relative-entropy-scaling of \mathbf{A} to (\mathbf{r}, \mathbf{c}) . Indeed, a generalized version of Pinsker's inequality provided in Lemma 13.2.1 shows that if $\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 \geq 5$, then $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \geq (1 - \ln 2) \cdot 4 > 1 \geq \delta$.¹⁰
- (ii) If $\gamma \leq 10$, then $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq 10(1 + \delta/80) \leq 15$ and a multiplicative $(1 \pm \delta/80)$ -approximation of $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ thus forms an additive $15\delta/80 \leq \delta/4$ -approximation of $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$.

Finally, to estimate the last term, note that an additive $\delta/4$ -approximation of $\ln(r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))/r_i)$ (the output of `LogSumExp`) for each $i \in [n]$ leads to an approximation of $\sum_{i=1}^n r_i \ln(\frac{r_i}{r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))})$ with additive error at most $\sum_{i=1}^n r_i \delta/4 = \delta/4$. Therefore, the quantity $\gamma - 1 + \sum_{\ell=1}^n r_\ell a_\ell$ computed in `TestScaling` is a $\delta/2$ -additive approximation of $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})))$. If $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \delta$, then the approximation is at most $3\delta/2$, and the condition evaluates to **True**. Similarly, if $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \geq 2\delta$, then the approximation is at least $3\delta/2$, and the condition evaluates to **False**. Note that if the approximation is exactly $3\delta/2$, then either return value is acceptable. We can compute a $\delta/2$ -additive approximation of $D(\mathbf{c} \parallel \mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})))$ in the same manner, showing that our implementation satisfies the requirements.

Finally, for the time complexity of `TestScaling`, note that each call to `LogSumExp` takes time $\tilde{O}(\sqrt{s_\ell/\delta})$ where s_ℓ is the number of potentially non-zero entries of the ℓ -th row or column (and we suppress a polylogarithmic dependence on n). Since we call `LogSumExp` once for each row and column and the square-root is a concave function, we thus obtain a time complexity of $\tilde{O}(\sqrt{mn}/\delta)$ for `TestScaling`. \square

A similar proof argument leads to the following proposition.

Proposition 13.3.4 (TestBalancing). *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with zeroes on the diagonal, each row and column containing at least one non-zero element and all non-zero entries at least $\mu > 0$. Let $\delta, \eta \in (0, 1)$ be rational numbers. Then there exists an algorithm `TestBalancing` that when given query access to $\mathbf{x} \in \mathbb{R}^n$ encoded in (b_1, b_2) -fixed-point-format and η, δ as input, determines with success probability $\geq 1 - \eta$ whether $\mathbf{A}(\mathbf{x})$ is δ - H^2 -balanced, and uses $\tilde{O}(\sqrt{mn}/\delta \log(1/\eta))$ queries and other gates, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1 and b_2 , and polylogarithmic factors in n, m , and δ .*

Proof. The only change compared to `TestScaling` is understanding the error one occurs in estimating the squared Hellinger distance via $(1 \pm \varepsilon)$ -multiplicative

¹⁰Note that naively applying Pinsker's inequality implies that whenever $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \delta$ then also $\|\mathbf{r} - \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 = O(\sqrt{\delta})$, which would indeed imply that $\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 = O(1)$ for all $\delta \leq 1$. However, we may only apply Pinsker's inequality to probability distributions, which $\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))$ need not be. In Lemma 13.2.1 we show a generalized version of Pinsker's inequality that says that $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \geq \|\mathbf{r} - \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 - \ln(1 + \|\mathbf{r} - \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1)$ (the latter is lower bounded by $\|\cdot\|_1^2/4$ on $[0, 1]$ and by $(1 - \ln 2)\|\cdot\|_1$ on $[1, \infty)$).

estimates \tilde{r}_i, \tilde{c}_i of the i -th row- and column marginals of $\mathbf{A}(\mathbf{x})$, and deducing a suitable ε . Observe that

$$\begin{aligned}
 H(\mathbf{r}, \mathbf{c})^2 &= \sum_i (\sqrt{r_i} - \sqrt{c_i})^2 \\
 &= \sum_i r_i + c_i - 2\sqrt{r_i c_i} \\
 &\leq \sum_i \tilde{r}_i / (1 - \varepsilon) + \tilde{c}_i / (1 - \varepsilon) - 2\sqrt{\tilde{r}_i \tilde{c}_i} (1 + \varepsilon) \\
 &= \sum_i \left((\tilde{r}_i + \tilde{c}_i) - 2\sqrt{\tilde{r}_i \tilde{c}_i} \right) / (1 - \varepsilon) - 2\sqrt{\tilde{r}_i \tilde{c}_i} (1 + \varepsilon - 1/(1 - \varepsilon)) \\
 &\leq \sum_i \left((\tilde{r}_i + \tilde{c}_i) - 2\sqrt{\tilde{r}_i \tilde{c}_i} \right) / (1 - \varepsilon) - 2(\tilde{r}_i + \tilde{c}_i) (1 + \varepsilon - 1/(1 - \varepsilon)).
 \end{aligned}$$

For $\varepsilon \leq \frac{1}{2}$ one has the estimate $-(1 + \varepsilon - 1/(1 - \varepsilon)) \leq \varepsilon$, and so we get $H(\mathbf{r}, \mathbf{c})^2 \leq H(\tilde{\mathbf{r}}, \tilde{\mathbf{c}})^2 / (1 - \varepsilon) + 2\varepsilon(\|\tilde{\mathbf{r}}\|_1 + \|\tilde{\mathbf{c}}\|_1)$. This estimate implies that we can guarantee $H(\mathbf{r}, \mathbf{c})^2 \leq \delta \|\mathbf{A}(\mathbf{x})\|_1$ by computing $H(\tilde{\mathbf{r}}, \tilde{\mathbf{c}})^2$, $\|\tilde{\mathbf{r}}\|_1$ and $\|\tilde{\mathbf{c}}\|_1$ and verifying that $H(\tilde{\mathbf{r}}, \tilde{\mathbf{c}})^2 / (1 - \varepsilon) + 2\varepsilon(\|\tilde{\mathbf{r}}\|_1 + \|\tilde{\mathbf{c}}\|_1) \leq \delta \|\mathbf{A}(\mathbf{x})\|_1$, for which it suffices to take $\varepsilon = \Theta(\delta)$. Similarly, $\varepsilon = \Theta(\delta)$ and an argument as above will also suffice to establish an inequality of the form $H(\mathbf{r}, \mathbf{c})^2 \geq \delta \|\mathbf{A}(\mathbf{x})\|_1$, i.e., this method allows one to certify that the matrix is far from balanced. \square

14. Quantum Sinkhorn and Osborne algorithms

In this chapter we present quantum algorithms for matrix scaling and balancing that are based on the well-known and natural Sinkhorn and Osborne algorithms. They can be thought of as first-order alternating minimization algorithms. In Section 14.1 we start by discussing a quantum version of Sinkhorn’s algorithm for matrix scaling, which we analyze in detail. We then describe two important variations: In Section 14.2, we give an improved analysis which shows that fewer iterations are required when scaling matrices that are entrywise positive, and in Section 14.3, we describe a quantum version of a randomized variant of the Sinkhorn algorithm, which has been of recent interest given its good performance in practice (but is more difficult to analyze in the quantum setting). Finally, in Section 14.4 we discuss a quantum version of a randomized variant of the Osborne algorithm for matrix balancing by drawing on similar ideas.

14.1. Quantum Sinkhorn algorithm

In this section we state Algorithm 14.1, a variant of the well-known Sinkhorn algorithm, and provide its analysis. The objective of Sinkhorn’s algorithm is to find scaling vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ such that the matrix $\mathbf{A}(\mathbf{x}, \mathbf{y}) = (A_{ij}e^{x_i+y_j})_{i,j \in [n]}$ has row and column marginals \mathbf{r} and \mathbf{c} , respectively. It does so in an iterative way. Starting from the rational matrix $\mathbf{A} \in [0, 1]^{n \times n}$, it finds a vector \mathbf{x} such that the row marginals of $(A_{ij}e^{x_i})$ are \mathbf{r} , and then it finds a \mathbf{y} such that the column marginals of $\mathbf{A}(\mathbf{x}, \mathbf{y})$ are \mathbf{c} . The second step may have changed the row marginals, so we repeat the procedure. We can view this as updating the coordinates of \mathbf{x} and \mathbf{y} one at a time, starting from the all-0 vectors. To update the row scaling vectors, we wish to find $\mathbf{x}' = \mathbf{x} + \Delta$ such that

$$\mathbf{r}(\mathbf{A}(\mathbf{x}', \mathbf{y})) = \mathbf{r}.$$

Expanding the above equation yields

$$e^{\Delta_\ell} \cdot r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) = r_\ell,$$

for $\ell \in [n]$. Since we assume every row and column contains at least one non-zero entry, the above equation has a unique solution, resulting in the following formula:

$$x'_\ell = x_\ell + \Delta_\ell = x_\ell + \ln \left(\frac{r_\ell}{r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))} \right) = \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j}} \right). \quad (14.1.1)$$

This chapter is adapted from [AGL+21].

14. Quantum Sinkhorn and Osborne algorithms

Analogously, we can achieve that $c(\mathbf{A}(\mathbf{x}, \mathbf{y}')) = \mathbf{c}$ if we instead update $\mathbf{y}' = \mathbf{y} + \Delta$, where

$$y'_\ell = y_\ell + \Delta_\ell = y_\ell + \ln \left(\frac{c_\ell}{c_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))} \right) = \ln \left(\frac{c_\ell}{\sum_{i=1}^n A_{i\ell} e^{x_i}} \right). \quad (14.1.2)$$

We use the term “one Sinkhorn iteration” to refer to the process of updating all n row scaling vectors, or updating all n column scaling vectors.

We study a version of Sinkhorn’s algorithm where, instead of computing row and column marginals in each iteration exactly, we use a *multiplicative* approximation of the marginals to compute δ -additive approximations of Eqs. (14.1.1) and (14.1.2). In the classical literature, the approximation errors can be chosen to be very small, since the cost per iteration scales as $\text{polylog}(1/\delta)$, and hence that error is essentially a minor technical detail. In the quantum setting, we can obtain a better dependence in terms of n at the cost of allowing for a $\text{poly}(1/\delta)$ -dependence. Therefore, in the analysis below we pay particular attention to the required precision δ . We state the Sinkhorn algorithm in terms of the two subroutines described in Section 13.3. For both subroutines we provided both classical and quantum implementations. For the analysis of Algorithm 14.1, we only use the guarantees of the subroutines as stated, and do not refer to their actual implementation.

The first subroutine we use is `LogSumExp`, which is used to update the scaling vectors. In odd iterations we update the row scaling vector \mathbf{x} according to Eq. (14.1.1), while in even iterations we update the column scaling factors \mathbf{y} according to Eq. (14.1.2) – in both cases with additive precision δ assuming the subroutine does not fail. The second subroutine is `TestScaling`, which tests whether scaling vectors (\mathbf{x}, \mathbf{y}) yield a relative entropy scaling of the desired precision. Both of these subroutines have a precision parameter and an upper bound on their failure probability. Note that allowing for the possibility of failure is essential since the quantum implementation of the subroutines is inherently probabilistic.

The Sinkhorn algorithm thus has a number of tunable parameters. We provide an upper bound T on the number of Sinkhorn iterations to be performed, and a choice of fixed-point format (b_1, b_2) , which is used for storing each entry of the scaling vectors (\mathbf{x}, \mathbf{y}) . Apart from that, we use two precision parameters $\delta, \delta' \in (0, 1)$, one for each subroutine used in the algorithm, and a failure probability $\eta \in [0, 1]$ for each individual subroutine call. In Proposition 14.1.7 we show how to choose these parameters for Algorithm 14.1 such that the output $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n$ forms an ε -relative-entropy-scaling of \mathbf{A} to (\mathbf{r}, \mathbf{c}) with probability at least $2/3$. The resulting time complexity for running the algorithm with these parameters gives us our main result of this section, Theorem 14.1.8, where we use results from Section 13.3 for the cost of implementing `LogSumExp` and `TestScaling`. Note that the error as measured in relative entropy can be converted to ℓ^1 -error using (a generalization of) Pinsker’s inequality (cf. Lemma 13.2.1).

Algorithm 14.1: Full Sinkhorn with finite precision and failure probability

Input: Query access to $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and non-zero entries at least $\mu > 0$, target marginals $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, iteration count $T \in \mathbb{N}$, bit counts $b_1, b_2 \in \mathbb{N}$, estimation precision $0 < \delta < 1$, test precision $0 < \delta' < 1$ and subroutine failure probability $\eta \in [0, 1]$

Output: Vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with entries encoded in (b_1, b_2) fixed-point format

Analysis: Theorems 14.1.8 and 14.2.5 and Corollaries 14.1.9 and 14.2.6

```

1  $\mathbf{x}^{(0)}, \mathbf{y}^{(0)} \leftarrow \mathbf{0}$ ; // entries in  $(b_1, b_2)$  fixed-point format
2 for  $t \leftarrow 1, 2, \dots, T$  do
3   if  $t$  is odd then
4     for  $\ell \leftarrow 1, 2, \dots, n$  do
5        $x_\ell^{(t)} \leftarrow -\text{LogSumExp}(\mathbf{A}_{\ell \bullet}, r_\ell, \mathbf{y}^{(t-1)}, \delta, b_1, b_2, \eta, \mu)$ ;
6     end for
7      $\mathbf{y}^{(t)} \leftarrow \mathbf{y}^{(t-1)}$ ;
8   else if  $t$  is even then
9     for  $\ell \leftarrow 1, 2, \dots, n$  do
10       $y_\ell^{(t)} \leftarrow -\text{LogSumExp}(\mathbf{A}_{\bullet \ell}, c_\ell, \mathbf{x}^{(t-1)}, \delta, b_1, b_2, \eta, \mu)$ ;
11    end for
12     $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)}$ ;
13  end if
14  if  $\text{TestScaling}(\mathbf{A}, \mathbf{r}, \mathbf{c}, \mathbf{x}^{(t)}, \mathbf{y}^{(t)}, \delta', b_1, b_2, \eta, \mu)$  then
15    return  $(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})$ ;
16  end if
17 end for
18 return  $(\mathbf{x}^{(T)}, \mathbf{y}^{(T)})$ ;

```

14.1.1. The potential for matrix scaling

The analysis will be based on a potential argument, using the following convex function (already mentioned in the introduction) as potential:

$$f(\mathbf{x}, \mathbf{y}) = \sum_{i,j=1}^n A_{ij} e^{x_i + y_j} - \sum_{i=1}^n r_i x_i - \sum_{j=1}^n c_j y_j.$$

This potential function is often used in the context of matrix scaling, as its gradient is precisely the difference between the current and the desired marginals (as we mentioned in Section 13.1). Many of the more sophisticated algorithms for matrix scaling also try to minimize this function directly, see Chapter 15. For our purposes, we first state a bound on the potential gap $f(\mathbf{0}, \mathbf{0}) - \inf_{\mathbf{x}, \mathbf{y} \in \mathbb{R}^n} f(\mathbf{x}, \mathbf{y})$, proven by relating it to the (shifted) Kempf–Ness function as in Chapters 2 and 5. For matrices \mathbf{A} that are *exactly* (\mathbf{r}, \mathbf{c}) -scalable, this bound is well-known (see e.g. [KLR07; CK21]), but to the best of our knowledge, it has not yet appeared in the literature when \mathbf{A} is only assumed to be asymptotically scalable to (\mathbf{r}, \mathbf{c}) .

14. Quantum Sinkhorn and Osborne algorithms

Theorem 14.1.1. Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ be a non-zero matrix with non-negative entries, and let $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{\geq 0}^n$ such that $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$. Then the following statements are equivalent:

(i) The function $f: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}, \mathbf{y}) = \sum_{i,j=1}^n A_{ij} e^{x_i + y_j} - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle$$

is bounded from below.

(ii) The function $F: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$F(\mathbf{x}, \mathbf{y}) = \ln \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right) - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle$$

is bounded from below.

(iii) The matrix \mathbf{A} is asymptotically (\mathbf{r}, \mathbf{c}) -scalable.

(iv) The point (\mathbf{r}, \mathbf{c}) is in the convex hull of the set

$$\Omega = \{ \boldsymbol{\omega}_{ij} = (\mathbf{e}_i, \mathbf{e}_j) : A_{ij} > 0 \} \subseteq \mathbb{R}^n \times \mathbb{R}^n.$$

Furthermore, if any of these conditions hold, then

$$f(\mathbf{0}, \mathbf{0}) - \inf_{\mathbf{x}, \mathbf{y} \in \mathbb{R}^n} f(\mathbf{x}, \mathbf{y}) \leq \|\mathbf{A}\|_1 - 1 + \ln(1/\mu) \quad (14.1.3)$$

where μ is the smallest non-zero entry of \mathbf{A} .

Proof. The equivalence of (ii), (iii) and (iv) follows directly from Proposition 5.1.2. It remains to establish the equivalence between (i) and (ii), and Eq. (14.1.3).

First, we claim that for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, we have the equality

$$\min_{t \in \mathbb{R}} f(\mathbf{x} + t\mathbf{1}, \mathbf{y}) = 1 + F(\mathbf{x}, \mathbf{y}).$$

Consider the function $g: \mathbb{R} \rightarrow \mathbb{R}$ given by $g(t) = f(\mathbf{x} + t\mathbf{1}, \mathbf{y})$. Then

$$\begin{aligned} g(t) &= \sum_{i,j=1}^n A_{ij} e^{x_i + y_j + t} - \langle \mathbf{r}, \mathbf{x} \rangle - t \langle \mathbf{r}, \mathbf{1} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle \\ &= \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right) e^t - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle - t \end{aligned}$$

since $\langle \mathbf{r}, \mathbf{1} \rangle = \|\mathbf{r}\|_1 = 1$. From this expression, it is clear that $g(t)$ is strictly convex, and attains its minimum at $t^* \in \mathbb{R}$ such that

$$g'(t^*) = \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right) e^{t^*} - 1 = 0,$$

i.e.,

$$t^* = -\ln \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right).$$

Consequently, we see that

$$\min_{t \in \mathbb{R}} g(t) = g(t^*) = 1 - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle + \ln \left(\sum_{i,j=1}^n A_{ij} e^{x_i + y_j} \right) = 1 + F(\mathbf{x}, \mathbf{y})$$

as desired.

To establish the potential bound, we observe that

$$\begin{aligned} f(\mathbf{0}, \mathbf{0}) - \inf_{(\mathbf{x}, \mathbf{y})} f(\mathbf{x}, \mathbf{y}) &= \|\mathbf{A}\|_1 - \inf_{(\mathbf{x}, \mathbf{y})} (1 + F(\mathbf{x}, \mathbf{y})) \\ &= \|\mathbf{A}\|_1 - 1 - \inf_{(\mathbf{x}, \mathbf{y})} F(\mathbf{x}, \mathbf{y}) \\ &\leq \|\mathbf{A}\|_1 - 1 - \ln(\mu) \\ &= \|\mathbf{A}\|_1 - 1 + \ln(1/\mu), \end{aligned}$$

where the first equality follows from the above claim and the inequality follows from Proposition 5.1.2. \square

14.1.2. Bounding the number of iterations

One can show that, for a Sinkhorn iteration in which we update the rows exactly, i.e., $\hat{x}_\ell = \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})$ for $\ell \in [n]$, the potential decreases by exactly the relative entropy:

$$f(\mathbf{x}, \mathbf{y}) - f(\hat{\mathbf{x}}, \mathbf{y}) = D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))), \quad (14.1.4)$$

and similarly for exact column updates. The next lemma generalizes this to allow for error in the update; it shows that we can lower bound the decrease of the potential function in every iteration in terms of the relative entropy between the target marginal and the current marginal, under the assumption that every call to the subroutine `LogSumExp` succeeds.

Lemma 14.1.2. *Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$, let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, let $\delta \in [0, 1]$, and let $\hat{\mathbf{x}} \in \mathbb{R}^n$ be a vector such that for every $\ell \in [n]$, we have $|\hat{x}_\ell - \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})| \leq \delta$. Then*

$$f(\mathbf{x}, \mathbf{y}) - f(\hat{\mathbf{x}}, \mathbf{y}) \geq D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) - 2\delta.$$

A similar statement holds for an update of \mathbf{y} (using \mathbf{c} instead of \mathbf{r} in the relative entropy).

Proof. We first note that we have the equalities

$$\begin{aligned} f(\mathbf{x}, \mathbf{y}) - f(\hat{\mathbf{x}}, \mathbf{y}) &= \sum_{\ell,j=1}^n A_{\ell j} e^{x_\ell + y_j} - \sum_{\ell,j=1}^n A_{\ell j} e^{\hat{x}_\ell + y_j} - \sum_{i=1}^n r_i \cdot (x_i - \hat{x}_i) \\ &= \sum_{\ell=1}^n (r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_\ell(\mathbf{A}(\hat{\mathbf{x}}, \mathbf{y})) - r_\ell \cdot (x_\ell - \hat{x}_\ell)) \end{aligned}$$

14. Quantum Sinkhorn and Osborne algorithms

Denote $z_\ell = \hat{x}_\ell - \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})$, so that $|z_\ell| \leq \delta$. Note that

$$r_\ell(\mathbf{A}(\hat{\mathbf{x}}, \mathbf{y})) = e^{\hat{x}_\ell} \sum_{j=1}^n A_{\ell j} e^{y_j} = r_\ell e^{z_\ell}.$$

Furthermore, we also have

$$x_\ell - \hat{x}_\ell = \ln\left(\frac{1}{r_\ell} \sum_{j=1}^n A_{\ell j} e^{x_\ell + y_j}\right) - z_\ell = -\ln\left(\frac{r_\ell}{r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))}\right) - z_\ell.$$

Therefore we can rewrite

$$r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_\ell(\mathbf{A}(\hat{\mathbf{x}}, \mathbf{y})) - r_\ell \cdot (x_\ell - \hat{x}_\ell) = r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_\ell(e^{z_\ell} - z_\ell) + r_\ell \ln\left(\frac{r_\ell}{r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))}\right).$$

For $z \in [-1, 1]$ one can easily show that $e^z - z \leq 1 + 2|z|$, and so

$$\begin{aligned} & r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_\ell(e^{z_\ell} - z_\ell) + r_\ell \ln\left(\frac{r_\ell}{r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))}\right) \\ & \geq r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_\ell - 2r_\ell|z_\ell| + r_\ell \ln\left(\frac{r_\ell}{r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))}\right) \\ & = \rho(r_\ell \| r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))) - 2r_\ell|z_\ell|, \end{aligned}$$

so that we may conclude

$$f(\mathbf{x}, \mathbf{y}) - f(\hat{\mathbf{x}}, \mathbf{y}) \geq D(\mathbf{r} \| \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) - 2 \sum_{\ell=1}^n r_\ell |z_\ell| \geq D(\mathbf{r} \| \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) - 2\delta$$

since $|z_\ell| \leq \delta$ for every $\ell \in [n]$, and $\|\mathbf{r}\|_1 = 1$. □

14.1.3. Controlling the bit complexity

The previous lemma showed that updating the scaling vectors with additive precision δ suffices to make progress in minimizing the potential function f , as long as we are still far away from the desired marginals (in relative entropy distance). As we wish to store the entries of \mathbf{x} and \mathbf{y} with additive precision $\delta > 0$ using a (b_1, b_2) fixed-point format, we need $b_2 \geq \lceil \log_2(1/\delta) \rceil$. The guarantees of `LogSumExp` and `TestScaling` assert that this choice of b_2 is also sufficient. Lemma 14.1.4 shows how large we need to take b_1 to ensure that the requirements of `LogSumExp` and `TestScaling` are satisfied in any particular iteration.

Remark 14.1.3. Note that the algorithm returns as soon as `TestScaling` returns `True`, or after T iterations. However, for the sake of simplifying the analysis, we always assume that $\mathbf{x}^{(t)}$ and $\mathbf{y}^{(t)}$ are defined for $t = 0, \dots, T$.

Lemma 14.1.4 (Bounding the scalings). *Let $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and non-zero entries at least $\mu > 0$. Let $T \geq 1$ and $\delta \in [0, 1]$. Denote $\sigma = \max(|\ln r_{\min}|, |\ln c_{\min}|)$. Let $b_2 = \lceil \log_2(1/\delta) \rceil$ and choose $b_1 = \lceil \log_2(T) + \log_2(\ln(\frac{1}{\mu}) + 1 + \sigma) \rceil$. If for all $t \in [T]$ the subroutine `LogSumExp` succeeds, then for all $t \in [T]$ and $\ell \in [n]$ we have*

$$\left| \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}}} \right) \right| \leq 2^{b_1}, \quad \left| \ln \left(\frac{c_\ell}{\sum_{i=1}^n A_{i\ell} e^{x_i^{(t)}}} \right) \right| \leq 2^{b_1}$$

and

$$\|(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})\|_\infty \leq t \left(\ln \left(\frac{1}{\mu} \right) + \delta + \sigma \right) \leq t \left(\ln \left(\frac{1}{\mu} \right) + 1 + \sigma \right).$$

Proof. We prove the norm bound by induction and the other claim as we go along. The norm bound clearly holds at time $t = 0$. Now assume it holds at time t . Assume that $t + 1$ is odd, so that we update the rows in iteration $t + 1$ (the case when $t + 1$ is even follows similarly). For each $\ell \in [n]$, we bound $x_\ell^{(t+1)}$. Note that, by assumption, we have

$$\delta \geq \left| x_\ell^{(t+1)} - \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}}} \right) \right| = \left| x_\ell^{(t+1)} - \ln(r_\ell) + \ln \left(\sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}} \right) \right|. \quad (14.1.5)$$

Observe that

$$-\ln \left(\sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}} \right) \geq -\ln \left(\sum_{j=1}^n A_{\ell j} e^{\|\mathbf{y}^{(t)}\|_\infty} \right) = -\|\mathbf{y}^{(t)}\|_\infty - \ln \left(\sum_{j=1}^n A_{\ell j} \right) \geq -\|\mathbf{y}^{(t)}\|_\infty,$$

where the last inequality uses $\sum_{j=1}^n A_{\ell j} \leq \|\mathbf{A}\|_1 \leq 1$. Similarly, for the upper bound, we have

$$-\ln \left(\sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}} \right) \leq -\ln \left(\sum_{j=1}^n A_{\ell j} e^{-\|\mathbf{y}^{(t)}\|_\infty} \right) \leq \|\mathbf{y}^{(t)}\|_\infty - \ln(\mu)$$

where we used that all non-zero entries of \mathbf{A} are at least μ , and every row contains at least one non-zero entry. Note that these bounds together with the choice of b_1 and the inductive assumption on $\|\mathbf{x}^{(t)}\|_\infty$ and $\|\mathbf{y}^{(t)}\|_\infty$ imply the first claim.

If we use these estimates in Eq. (14.1.5), we obtain

$$\ln(r_\ell) - \|\mathbf{y}^{(t)}\|_\infty - \delta \leq x_\ell^{(t+1)} \leq \|\mathbf{y}^{(t)}\|_\infty + \ln(r_\ell) + \ln \left(\frac{1}{\mu} \right) + \delta.$$

Since $\mu \leq 1$ and $r_\ell \leq 1$, this implies

$$|x_\ell^{(t+1)}| \leq \|\mathbf{y}^{(t)}\|_\infty + \ln \left(\frac{1}{\mu} \right) + |\ln(r_\ell)| + \delta.$$

This shows that $\|\mathbf{x}^{(t+1)}\|_\infty \leq \|\mathbf{y}^{(t)}\|_\infty + \ln \left(\frac{1}{\mu} \right) + |\ln(r_{\min})| + \delta$. The case that we updated the columns in the $(t + 1)$ -st iteration is treated completely similarly. Thus, we conclude that

$$\|(\mathbf{x}^{(t+1)}, \mathbf{y}^{(t+1)})\|_\infty \leq \|(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})\|_\infty + \ln \left(\frac{1}{\mu} \right) + \sigma + \delta.$$

By the induction hypothesis and $\delta \leq 1$, the desired upper bound holds at time $t + 1$, and it suffices to use b_1 bits in any iteration to meet the requirements of `LogSumExp`. \square

14.1.4. Analysis of quantum Sinkhorn

To formally analyze the expected progress it will be convenient to define the following events.

Definition 14.1.5 (Important events). For $t = 1, \dots, T$, we define the following events:

- Let S_t denote the event that all n calls to `LogSumExp` succeed in the t -th iteration.
- Define S to be the intersection of the events S_t , i.e., $S = \bigcap_{t=1}^T S_t$.

To give some intuition, we show below that the event S is the ‘good’ event where a row-update makes the relative entropy between \mathbf{r} and the updated row-marginals at most δ (and similarly for the columns). We only use Lemma 14.1.6 in Section 14.2.

Lemma 14.1.6. *If S holds and $\delta \leq 1$, then the following holds for all $t \in [T]$:*

- If t is odd, then $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))) \leq \delta$.
- If t is even, then $D(\mathbf{c} \parallel \mathbf{c}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))) \leq \delta$.

Proof. If S_t holds and t is odd, then

$$\left| x_\ell^{(t)} - \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j^{(t-1)}}} \right) \right| \leq \delta$$

for all $\ell \in [n]$, while $\mathbf{y}^{(t)} = \mathbf{y}^{(t-1)}$. Accordingly,

$$r_\ell(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})) = e^{x_\ell^{(t)}} \sum_{j=1}^n A_{\ell j} e^{y_j^{(t)}} \in [e^{-\delta} r_\ell, e^\delta r_\ell].$$

Since

$$\rho(r_\ell \parallel r_\ell e^z) = r_\ell e^z - r_\ell + r_\ell \ln \left(\frac{r_\ell}{r_\ell e^z} \right) = r_\ell (e^z - 1 - z) \leq r_\ell |z|$$

for any $|z| \leq 1$, we obtain

$$D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))) = \sum_{\ell=1}^n \rho(r_\ell \parallel r_\ell(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))) \leq \sum_{\ell=1}^n r_\ell \delta = \delta,$$

again using $\|\mathbf{r}\|_1 = 1$. A similar computation yields the result for even t . \square

We can combine Theorem 14.1.1 and Lemma 14.1.2 to show Algorithm 14.1 returns, with high probability, an ε -relative-entropy-scaling to (\mathbf{r}, \mathbf{c}) by choosing $\delta = O(\varepsilon)$.

Proposition 14.1.7. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and non-zero entries at least $\mu > 0$ and let $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$. Assume \mathbf{A} is asymptotically scalable to (\mathbf{r}, \mathbf{c}) . For $\varepsilon \in (0, 1]$, choose*

$$T = \left\lceil \frac{8}{\varepsilon} \ln \left(\frac{1}{\mu} \right) \right\rceil + 1,$$

$\delta = \frac{\varepsilon}{16}$, $\delta' = \frac{\varepsilon}{2}$, $\eta = \frac{1}{3(n+1)T}$, $b_2 = \lceil \log_2(\frac{1}{\delta}) \rceil$, and $b_1 = \lceil \log_2(T) + \log_2(\ln(\frac{1}{\mu}) + \sigma + 1) \rceil$, where $\sigma = \max(|\ln r_{\min}|, |\ln c_{\min}|)$. Then, Algorithm 14.1 with these parameters returns a pair (\mathbf{x}, \mathbf{y}) such that $D(\mathbf{r} \parallel \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \varepsilon$ and $D(\mathbf{c} \parallel \mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \varepsilon$ with probability $\geq 2/3$.

Proof. The choice of η is such that with probability at least $1 - (n+1)T\eta = 2/3$, the event S holds (i.e., all calls to `LogSumExp` succeed) and all calls to `TestScaling` succeed. Assume this is the case. If there exists an iteration $t \in [T]$ for which `TestScaling` outputs **True**, then we have obtained a $2\delta'$ -relative-entropy-scaling of \mathbf{A} to (\mathbf{r}, \mathbf{c}) , which is an ε -relative-entropy-scaling by the choice of δ' . We now bound the number of iterations in which `TestScaling` can output **False**, i.e., the number of iterations in which (\mathbf{x}, \mathbf{y}) is not a δ' -relative-entropy-scaling of \mathbf{A} . Suppose that $\tau \in [T]$ is the last iteration such that `TestScaling` outputs **False**. By Theorem 14.1.1, we have

$$f_0 - f_\tau \leq \ln \left(\frac{1}{\mu} \right),$$

which is positive since $\mu \leq 1$. We now lower bound the left-hand side by a telescoping sum: using Lemma 14.1.2 (and implicitly Lemma 14.1.4) we obtain

$$f_0 - f_\tau = \sum_{t=1}^{\tau} (f_{t-1} - f_t) \geq 2\tau\delta = \frac{\varepsilon\tau}{8}$$

It follows that $\tau \leq 8 \ln(\frac{1}{\mu})/\varepsilon < T$, so `TestScaling` must output **True** in the T -th iteration at the latest. This concludes the proof. \square

With the performance guarantees provided by Theorems 13.3.2 and 13.3.3 for quantum implementations of `LogSumExp` and `TestScaling`, we can state the time complexity of computing an ε -relative-entropy-scaling of \mathbf{A} to marginals (\mathbf{r}, \mathbf{c}) . We now prove the main theorem of this section, already stated earlier and repeated here for convenience.

Theorem 14.1.8. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with $\|\mathbf{A}\|_1 \leq 1$ and m non-zero entries, each at least $\mu > 0$, let $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, and let $\varepsilon \in (0, 1]$. Assume \mathbf{A} is asymptotically scalable to (\mathbf{r}, \mathbf{c}) . Then there exists a quantum algorithm (Algorithm 14.1 with parameters chosen as in Proposition 14.1.7) that, given sparse query access to \mathbf{A} , with probability $\geq 2/3$, computes $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n$ such that $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is ε -relative-entropy-scaled to (\mathbf{r}, \mathbf{c}) , for a total time complexity of $\tilde{O}(\sqrt{mn}/\varepsilon^{1.5})$.*

Proof. We show that Algorithm 14.1 with parameters chosen as in Proposition 14.1.7 has the stated time complexity. Note that the cost of computing these parameters from the input will be dominated by the runtime of the algorithm. Proposition 14.1.7 shows that Algorithm 14.1 runs for at most $O(\ln(1/\mu)/\varepsilon)$ iterations. Next we show

the time complexity per iteration is $\tilde{O}(\sqrt{mn}/\varepsilon)$, which implies the claimed total time complexity of $\tilde{O}(\sqrt{mn}/\varepsilon^{1.5})$.

Theorem 13.3.2 shows that invoking `LogSumExp` with precision $\delta = \Theta(\varepsilon)$ on a row containing s potentially non-zero entries incurs a cost of order $\tilde{O}(\sqrt{s}/\varepsilon)$, where we suppress a polylogarithmic dependence on n . Since in one iteration of Algorithm 14.1 we apply `LogSumExp` once to each row or once to each column, using Cauchy–Schwarz the total cost of the calls to `LogSumExp` in one iteration is

$$\tilde{O}\left(\sum_{i=1}^n \sqrt{s_i^r/\varepsilon} + \sum_{j=1}^n \sqrt{s_j^c/\varepsilon}\right) \subseteq \tilde{O}\left(\sqrt{mn}/\varepsilon\right),$$

where we recall that s_i^r and s_j^c are the numbers of potentially non-zero entries in the i -th row and j -th column of \mathbf{A} , respectively, and m is the total number of potentially non-zero entries in \mathbf{A} (i.e., $\sum_{i=1}^n s_i^r = m = \sum_{j=1}^n s_j^c$). Similarly, Theorem 13.3.3 shows invoking `TestScaling` with precision $\delta' = \Theta(\varepsilon)$ incurs a cost of order $\tilde{O}(\sqrt{mn}/\varepsilon)$. Finally we observe that compiling the quantum circuits (and preparing their inputs) for the calls to `LogSumExp` and `TestScaling` can be done with at most a polylogarithmic overhead. \square

Note that the dependency on $\ln(1/\mu)$ is suppressed by the $\tilde{O}(\cdot)$, since we assume that the numerator and denominator of any rational number in the input is bounded above by a polynomial in n . Using a generalization of Pinsker’s inequality (cf. Lemma 13.2.1), an ε -relative-entropy-scaling is a $O(\sqrt{\varepsilon})$ - ℓ^1 -scaling, which implies the following:

Corollary 14.1.9. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with $\|\mathbf{A}\|_1 \leq 1$ and m non-zero entries, each at least $\mu > 0$, let $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, and let $\varepsilon \in (0, 1]$. Assume \mathbf{A} is asymptotically scalable to (\mathbf{r}, \mathbf{c}) . Then there exists a quantum algorithm that, given sparse query access to \mathbf{A} , with probability $\geq 2/3$, computes $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n$ such that $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is ε - ℓ^1 -scaled to (\mathbf{r}, \mathbf{c}) , at a total time complexity of $\tilde{O}(\sqrt{mn}/\varepsilon^3)$.*

In Chapter 16 we show that the complexity in terms of n and m is tight (up to logarithmic factors): $\Omega(\sqrt{mn})$ queries to the input are needed to solve the scaling problem for constant ℓ^1 -error (by Pinsker’s inequality, the same lower bound is then implied for relative-entropy scaling as well).

14.2. Improved analysis for entrywise-positive matrices

We now show that if the matrix \mathbf{A} is entrywise positive, then the number of iterations to obtain an ε -relative-entropy-scaling reduces from $\tilde{O}(1/\varepsilon)$ to $\tilde{O}(1/\sqrt{\varepsilon})$, leading to a quantum algorithm with time complexity $\tilde{O}(n^{1.5}/\varepsilon)$. This also implies that one can find an ε - ℓ^1 -scaling in time $\tilde{O}(n^{1.5}/\varepsilon^2)$. These results, which are stated as Theorem 14.2.5 and Corollary 14.2.6, improve over Theorem 14.1.8 and Corollary 14.1.9. In this section, $\tilde{O}(\cdot)$ always suppresses polylogarithmic factors in n and $1/\varepsilon$.

The argument follows [KLR07], where a similar result was proved for ℓ^2 -scaling, and [DGK18] where it is extended to ℓ^1 -scaling (but implicitly by bounding the

relative entropy, as we do below). Our contribution is to show that their analyses are robust with respect to only using estimates of marginals, and observing that it extends to the relative-entropy setting. The key idea is the following: for entrywise-positive matrices, the scaling vectors \mathbf{x}, \mathbf{y} produced by the Sinkhorn algorithm each have *variation norm* ($x_{\max} - x_{\min}$) bounded by a *constant*, whereas for arbitrary matrices we can only show a bound that is linear in the number of iterations (Lemma 14.1.4). We state it below in Lemma 14.2.1. This is a variant of [KLR07, Lem. 6.2] This is also the only part of the improved analysis which requires entrywise positivity of \mathbf{A} .

Lemma 14.2.1. *Let $0 < \mu < \nu \leq 1$ and assume $\mathbf{A} \in [\mu, \nu]^{n \times n}$ and $\mathbf{r} \in \mathbb{R}_{>0}^n$ strictly positive. Let $\mathbf{y} \in \mathbb{R}^n$, let $\delta \geq 0$, and let $\mathbf{x}' \in \mathbb{R}^n$ be such that $|x'_\ell - \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})| \leq \delta$ for all $\ell \in [n]$. Then*

$$x'_{\max} - x'_{\min} \leq 2\delta + \ln \frac{\nu}{\mu} + \ln \frac{r_{\max}}{r_{\min}}.$$

An analogous statement holds for the column-scaling vectors after a column update. As a consequence, if $(\mathbf{x}^, \mathbf{y}^*) \in \mathbb{R}^n \times \mathbb{R}^n$ is such that $\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)$ is exactly (\mathbf{r}, \mathbf{c}) -scaled, then*

$$x^*_{\max} - x^*_{\min} \leq \ln \frac{\nu}{\mu} + \ln \frac{r_{\max}}{r_{\min}}, \quad y^*_{\max} - y^*_{\min} \leq \ln \frac{\nu}{\mu} + \ln \frac{c_{\max}}{c_{\min}}.$$

In other words, the variation in the row-scaling vectors after a δ -approximate Sinkhorn update is bounded above by 2δ plus a quantity depending only on \mathbf{A} and \mathbf{r} .

Proof. For any $\ell \in [n]$, we have $|x'_\ell - \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})| \leq \delta$. By using the upper and lower bound on the entries of \mathbf{A} , we obtain

$$\begin{aligned} & \ln \left(\frac{r_{\max}}{\mu \sum_{j=1}^n e^{y_j}} \right) + \delta \geq \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j}} \right) + \delta \\ & \geq x'_\ell \geq \ln \left(\frac{r_\ell}{\sum_{j=1}^n A_{\ell j} e^{y_j}} \right) - \delta \geq \ln \left(\frac{r_{\min}}{\nu \sum_{j=1}^n e^{y_j}} \right) - \delta. \end{aligned}$$

Therefore, for any $k, \ell \in [n]$, we obtain

$$x'_k - x'_\ell \leq \ln \left(\frac{r_{\max}}{\mu \sum_{j=1}^n e^{y_j}} \right) + \delta - \ln \left(\frac{r_{\min}}{\nu \sum_{j=1}^n e^{y_j}} \right) + \delta = 2\delta + \ln \frac{r_{\max}}{r_{\min}} + \ln \frac{\nu}{\mu}$$

as desired.

Assume now that $(\mathbf{x}^*, \mathbf{y}^*)$ exactly scale \mathbf{A} to (\mathbf{r}, \mathbf{c}) . Then an exact full Sinkhorn update does not change the scaling vectors, so it holds that $x^*_\ell = \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j^*})$ for all $\ell \in [n]$. We may thus apply the above obtained bound with $\delta = 0$, which provides the desired bound on the variation norm of \mathbf{x}^* . The bound on the variation norm of \mathbf{y}^* is proved completely analogously. \square

Note that the above proof fails if \mathbf{A} does not have full support; one can still attempt to use an upper and lower bound on the non-zero entries of \mathbf{A} , but the support in the k -th and ℓ -th rows generally differ, so the corresponding (logarithms of) sums of column-scaling vectors do not necessarily cancel.

The convexity of the potential f can then be used, along with the previous fact, to determine a potential bound which becomes better as the scaling error goes down.

14. Quantum Sinkhorn and Osborne algorithms

Lemma 14.2.2. *Let $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n$ and $\delta \in [0, 1/2]$ be such that*

$$|\mathbf{y}_e - \ln(\mathbf{c}_e / \sum_{i=1}^n A_{ie} e^{x_i})| \leq \delta.$$

Then, for any $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathbb{R}^n \times \mathbb{R}^n$,

$$f(\mathbf{x}, \mathbf{y}) - f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \leq \delta + (\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 + 2\delta)(x_{\max} - x_{\min} + \tilde{x}_{\max} - \tilde{x}_{\min}).$$

This lemma can be viewed as a robust version (i.e., it allows for finite precision in the Sinkhorn updates) of [DGK18, Lem. 2] and [KLRS07, Lem. 6.1, Lem. 6.2] (in the ℓ^2 -setting). We first prove Lemma 14.2.2 for $\delta = 0$, yielding an ℓ^1 -analog of [KLRS07, Lem. 6.1].

Lemma 14.2.3. *Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ and let $(\mathbf{x}, \mathbf{y}), (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathbb{R}^n \times \mathbb{R}^n$. If (\mathbf{x}, \mathbf{y}) is such that $\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{c}$, then*

$$f(\mathbf{x}, \mathbf{y}) - f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \leq \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1(x_{\max} - x_{\min} + \tilde{x}_{\max} - \tilde{x}_{\min}).$$

A similar statement holds if $\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{r}$.

Proof. We have

$$\begin{aligned} \text{grad}_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) &= \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}, \\ \text{grad}_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) &= \mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{c}, \end{aligned}$$

so in particular, if $\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{c}$, then $\text{grad}_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) = 0$. Now, by convexity of f , we have

$$f(\mathbf{x}, \mathbf{y}) + \langle \text{grad}_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}), \tilde{\mathbf{x}} - \mathbf{x} \rangle \leq f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}),$$

which we rearrange as

$$f(\mathbf{x}, \mathbf{y}) - f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \leq \langle \text{grad}_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}), \mathbf{x} - \tilde{\mathbf{x}} \rangle = \langle \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}, \mathbf{x} - \tilde{\mathbf{x}} \rangle. \quad (14.2.1)$$

Since $\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y})) = \mathbf{c}$ and $\|\mathbf{c}\|_1 = 1$, we have $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 = 1$ and $\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 = 1$ as well. In particular,

$$\langle \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}, \mathbf{1} \rangle = 0.$$

Set $\mathbf{z} = \mathbf{x} - \frac{\langle \mathbf{x}, \mathbf{1} \rangle}{n} \mathbf{1}$ and $\tilde{\mathbf{z}} = \tilde{\mathbf{x}} - \frac{\langle \tilde{\mathbf{x}}, \mathbf{1} \rangle}{n} \mathbf{1}$. Then, using Eq. (14.2.1),

$$\begin{aligned} f(\mathbf{x}, \mathbf{y}) - f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) &\leq \langle \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}, \mathbf{x} - \tilde{\mathbf{x}} \rangle \\ &= \langle \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}, \mathbf{z} - \tilde{\mathbf{z}} \rangle \\ &\leq \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 (\|\mathbf{z}\|_{\infty} + \|\tilde{\mathbf{z}}\|_{\infty}) \\ &\leq \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 (z_{\max} - z_{\min} + \tilde{z}_{\max} - \tilde{z}_{\min}) \\ &= \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 (x_{\max} - x_{\min} + \tilde{x}_{\max} - \tilde{x}_{\min}) \end{aligned}$$

as desired. The last inequality holds because the entries of \mathbf{z} and of $\tilde{\mathbf{z}}$ sum to zero. \square

To deal with updates with finite precision, we adapt Lemma 14.2.3 to the case where $\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))$ and \mathbf{c} are only approximately equal.

Proof of Lemma 14.2.2. Let \mathbf{y}' be the vector defined by

$$y'_\ell = \ln \left(\frac{c_\ell}{\sum_{i=1}^n A_{i\ell} e^{x_i}} \right),$$

i.e., \mathbf{y}' is the vector of column-scaling vectors after an exact Sinkhorn column update starting from (\mathbf{x}, \mathbf{y}) . Then

$$f(\mathbf{x}, \mathbf{y}) - f(\mathbf{x}, \mathbf{y}') = D(c \| c(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \leq \delta, \quad (14.2.2)$$

where the first inequality is Eq. (14.1.4) and the second inequality follows from the assumption on \mathbf{y} (see proof of Lemma 14.1.6; the event S_t corresponds precisely to the assumption on \mathbf{y}).

Furthermore, $c(\mathbf{A}(\mathbf{x}, \mathbf{y}')) = \mathbf{c}$, so we may apply Lemma 14.2.3 with $(\mathbf{x}, \mathbf{y}')$ and $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ to obtain

$$f(\mathbf{x}, \mathbf{y}') - f(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \leq \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}')) - \mathbf{r}\|_1 (x_{\max} - x_{\min} + \tilde{x}_{\max} - \tilde{x}_{\min}). \quad (14.2.3)$$

Since $y'_\ell - y_\ell \in [-\delta, \delta]$ for every $\ell \in [n]$, for every $i, j \in [n]$ we have

$$A_{ij} e^{x_i + y_j} \in [e^{-\delta} A_{ij} e^{x_i + y'_j}, e^{\delta} A_{ij} e^{x_i + y'_j}].$$

Since $\delta \leq 1/2$, we can use the estimates $e^{-\delta} \geq 1 - 2\delta$ and $e^{\delta} \leq 1 + 2\delta$, which imply that

$$r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y})) \in [(1 - 2\delta)r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}')), (1 + 2\delta)r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}'))]$$

for every $\ell \in [n]$. By the triangle inequality we get

$$\begin{aligned} \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}')) - \mathbf{r}\|_1 &\leq \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}')) - \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))\|_1 + \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 \\ &\leq 2\delta \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}'))\|_1 + \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1 \\ &= 2\delta + \|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) - \mathbf{r}\|_1. \end{aligned}$$

where the last equality holds since $\|\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}'))\|_1 = \|\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}'))\|_1 = \|\mathbf{c}\|_1 = 1$. If we plug this into Eq. (14.2.3) then together with Eq. (14.2.2) the proof is complete. \square

Together, Lemmas 14.2.1 and 14.2.2 yield the following adaptive bound on the potential gap for iterations produced by the Sinkhorn algorithm.

Corollary 14.2.4 (Adaptive potential gap bound). *Let $A \in [\mu, \nu]^{n \times n}$, let $t \geq 1$, and let $\mathbf{x}^{(t)}$ and $\mathbf{y}^{(t)}$ be as in Algorithm 14.1, and assume no call to `LogSumExp` has failed. If t is even, then we have*

$$f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) - f^* \leq \delta + 2 \left(\|\mathbf{r}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) - \mathbf{r}\|_1 + 2\delta \right) \left(\delta + \ln \frac{r_{\max}}{r_{\min}} + \ln \frac{\nu}{\mu} \right),$$

where $f^* = \min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y})$. A similar statement holds if t is odd.

Proof. It is well-known that any entrywise-positive matrix is exactly scalable to arbitrary strictly positive (\mathbf{r}, \mathbf{c}) . Thus, there exists $(\mathbf{x}^*, \mathbf{y}^*)$ such that $\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)$ is exactly (\mathbf{r}, \mathbf{c}) -scaled. This implies that $\nabla f(\mathbf{x}^*, \mathbf{y}^*) = 0$ and thus $f(\mathbf{x}^*, \mathbf{y}^*) = f^*$. By Lemma 14.2.1, we have $x_{\max}^* - x_{\min}^* \leq \ln \frac{r_{\max}}{r_{\min}} + \ln \frac{\nu}{\mu}$. Furthermore, as we assume no call to `LogSumExp` fails, we also have, by the same Lemma 14.2.1,

$$x_{\max}^{(t)} - x_{\min}^{(t)} \leq 2\delta + \ln \frac{r_{\max}}{r_{\min}} + \ln \frac{\nu}{\mu}.$$

The result now follows from applying Lemma 14.2.2 and a simple estimate. \square

14. Quantum Sinkhorn and Osborne algorithms

Note that Corollary 14.2.4 is stated in terms of the ℓ^1 -distance, which we can further upper bound in terms of the relative entropy using Pinsker's inequality. In this way, one can show that once \mathbf{A} is ε -relative-entropy-scaled for $\varepsilon \leq 1$, it takes $\tilde{O}(1/\sqrt{\varepsilon})$ full Sinkhorn iterations to obtain an $\varepsilon/2$ -relative-entropy-scaling. Obtaining an $O(1)$ -relative-entropy-scaling takes a constant number of Sinkhorn iterations, and from there onwards it suffices to halve the scaling error at most $\log_2(1/\varepsilon)$ times, where the number of iterations required to halve the scaling error increases by a factor $\sqrt{2}$ every time. Carefully keeping track of the total number of iterations then gives a total iteration count of $\tilde{O}(1/\sqrt{\varepsilon})$:

Theorem 14.2.5. *Let $0 < \mu < \nu \leq 1$, let $\mathbf{A} \in [\mu, \nu]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$, and let $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$. For $\varepsilon \in (0, 1]$, choose*

$$T = \left\lceil \frac{32 \ln(1/\mu) + \log_2(2/\varepsilon)(1 + 34C)}{\sqrt{\varepsilon}} \right\rceil,$$

$\delta = \varepsilon/64$, $\delta' = \varepsilon/2$, $\eta = 1/(3(n+1)T)$, $b_1 = \lceil \log_2(T) + \log_2(\ln(\frac{1}{\mu}) + 1 + \sigma) \rceil$, $b_2 = \lceil \log_2(1/\delta) \rceil$, where $C = \delta + \ln(r_{\max}/r_{\min}) + \ln(c_{\max}/c_{\min}) + \ln(\nu/\mu)$ and $\sigma = \max\{|\ln r_{\min}|, |\ln c_{\min}|\}$. Then, Algorithm 14.1 with these parameters returns a pair (\mathbf{x}, \mathbf{y}) such that, with probability $\geq 2/3$, $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is ε -relative-entropy-scaled to (\mathbf{r}, \mathbf{c}) . The resulting quantum algorithm has time complexity $\tilde{O}(n^{1.5}/\varepsilon^{1.5})$.

Proof. Observe first that we have chosen b_1, b_2 such that the guarantees of `LogSumExp` and `TestScaling` are satisfied at any iteration (cf. Lemma 14.1.4). Throughout the proof, we assume that all calls to `LogSumExp` and to `TestScaling` made by Algorithm 14.1 succeed, which by the choice of η happens with probability $\geq 2/3$. As always, we write $f^* = \inf_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y})$, and we abbreviate $f_t = f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})$, $\mathbf{r}^{(t)} = \mathbf{r}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))$, and $\mathbf{c}^{(t)} = \mathbf{c}(\mathbf{A}(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}))$ for the potential and the row and column marginals after the t -th iteration.

The strategy is as follows. For $t \geq 0$, define $\varepsilon_t = D(\mathbf{r} \parallel \mathbf{r}^{(t)})$ if t is even, and $\varepsilon_t = D(\mathbf{c} \parallel \mathbf{c}^{(t)})$ if t is odd. By Lemma 14.1.6, the other of the two relative entropies is at most δ for every $t \geq 1$, and so it suffices to bound the time until ε_t is sufficiently small. We will first bound the number of iterations until $\varepsilon_t \leq 1 - \ln 2$, and subsequently bound the number of iterations required for ε_t to halve.

We first argue that there exists an $N \leq 32 \ln(1/\mu)$ such that $\varepsilon_N \leq 1 - \ln 2$. Suppose for contradiction that this is not the case. Then for $t = 0, \dots, \lfloor 32 \ln(1/\mu) \rfloor$, we have $\varepsilon_t > 1 - \ln 2 \geq 1/4$, and by Theorem 14.1.1 and Lemma 14.1.2, we see that

$$\begin{aligned} \ln(1/\mu) &\geq f_0 - f^* \geq f_0 - f_{\lfloor 32 \ln(1/\mu) \rfloor + 1} = \sum_{t=0}^{\lfloor 32 \ln(1/\mu) \rfloor} (f_t - f_{t+1}) \geq \sum_{t=0}^{\lfloor 32 \ln(1/\mu) \rfloor} (\varepsilon_t - 2\delta) \\ &> (\lfloor 32 \ln(1/\mu) \rfloor + 1) ((1 - \ln 2) - 2\delta) \geq 32 \ln(1/\mu) \cdot \frac{1}{16} = 2 \ln(1/\mu) \end{aligned}$$

where we used $2\delta \leq 1/32$. This is the desired contradiction.

We now bound the halving time. Let $t \geq 1$ be such that $\varepsilon_t \leq 1 - \ln 2$, and define

$$N_t = \inf\{\tau \geq 0 : \varepsilon_{t+\tau} \leq \varepsilon_t/2\}. \quad (14.2.4)$$

For $\tau = 0, \dots, N_t - 1$, we have $\varepsilon_{t+\tau} > \varepsilon_t/2$, and so

$$f_t - f^* \geq f_t - f_{N_t} \geq \sum_{\tau=0}^{N_t-1} (f_{t+\tau} - f_{t+\tau+1}) \geq \sum_{\tau=0}^{N_t-1} (\varepsilon_{t+\tau} - 2\delta) \geq N_t \left(\frac{\varepsilon_t}{2} - 2\delta \right).$$

again by Theorem 14.1.1 and Lemma 14.1.2. Therefore, so long as $\varepsilon_t > 4\delta$, we obtain

$$N_t \leq \frac{f_t - f^*}{\varepsilon_t/2 - 2\delta}. \quad (14.2.5)$$

We first prove a bound on N_t assuming $8\delta \leq \varepsilon_t \leq 1 - \ln 2$. For t even, Lemma 13.2.1 then implies that $\|\mathbf{r} - \mathbf{r}^{(t)}\|_1 \leq 1$ (since $\varepsilon_t = D(\mathbf{r} \|\mathbf{r}^{(t)}) \leq 1 - \ln 2$, while the function $w(\alpha)$ is strictly larger than $1 - \ln 2$ for $\alpha > 1$) and hence $D(\mathbf{r} \|\mathbf{r}^{(t)}) \geq \|\mathbf{r} - \mathbf{r}^{(t)}\|_1^2/4$, so

$$\begin{aligned} N_t &\leq \frac{f_t - f^*}{\varepsilon_t/2 - 2\delta} \leq \frac{\delta + 2 \left(\|\mathbf{r}^{(t)} - \mathbf{r}\|_1 + 2\delta \right) C}{\varepsilon_t/2 - 2\delta} \\ &\leq \frac{\delta + 4(\sqrt{\varepsilon_t} + \delta) C}{\varepsilon_t/2 - 2\delta} \\ &= \frac{2\delta(1 + 4C) + 8\sqrt{\varepsilon_t}C}{\varepsilon_t - 4\delta} \\ &\leq \frac{(\varepsilon_t/4)(1 + 4C) + 8\sqrt{\varepsilon_t}C}{\varepsilon_t/2} = \frac{1 + 4C}{2} + \frac{16C}{\sqrt{\varepsilon_t}}, \end{aligned}$$

where the first inequality is Eq. (14.2.5), the second follows from Corollary 14.2.4, and in the last inequality we assume that $\varepsilon_t \geq 8\delta$. The same inequality holds for t odd, with an analogous proof. Thus, we have proved that, for any t such that $8\delta \leq \varepsilon_t \leq 1 - \ln 2$,

$$N_t \leq \frac{1 + 4C}{2} + \frac{16C}{\sqrt{\varepsilon_t}}. \quad (14.2.6)$$

We now combine the preceding to verify that the desired number of iterations suffices for Algorithm 14.1 to return an ε -relative-entropy-scaling. For $s \geq 0$, define

$$h_s = \min \left\{ t \geq 0 : \varepsilon_t \leq \frac{1 - \ln 2}{2^s} \right\}.$$

We proved above that $h_0 \leq 32 \ln(1/\mu)$. Clearly, the sequence h_s is non-decreasing. Let

$$S = \min \left\{ s \geq 0 : \frac{1 - \ln 2}{2^s} \leq 32\delta = \delta' \right\}.$$

Note that the algorithm will necessarily return within the first h_S iterations with an ε -relative-entropy-scaling. Indeed, either `TestScaling` returns `True` during one of the first $h_S - 1$ iterations, or it must return `True` in the h_S -th iteration, since then $\varepsilon_{h_S} \leq \delta'$ (and the other relative entropy is always at most $\delta \leq \delta'$). Thus it suffices to bound h_S . For any $0 \leq s < S$, if $h_{s+1} > h_s$ we have $\varepsilon_{h_s} > \frac{1 - \ln 2}{2^{s+1}} > 16\delta > 8\delta$, so

$$h_{s+1} - h_s \leq N_{h_s} \leq \frac{1 + 4C}{2} + \frac{16C}{\sqrt{\varepsilon_{h_s}}} < \frac{1 + 4C}{2} + 2^{s/2} \frac{16\sqrt{2}C}{\sqrt{1 - \ln 2}} \quad (14.2.7)$$

where the first inequality holds by definition of N_t in Eq. (14.2.4) and the second inequality is Eq. (14.2.6); the latter is applicable since $8\delta \leq \varepsilon_{h_s} \leq 1 - \ln 2$. Clearly Eq. (14.2.7) also holds if $h_{s+1} = h_s$. Thus we can upper bound the total number of iterations required by

$$\begin{aligned}
 h_S &= h_0 + \sum_{s=0}^{S-1} (h_{s+1} - h_s) \\
 &< 32 \ln(1/\mu) + \sum_{s=0}^{S-1} \left(\frac{1+4C}{2} + 2^{s/2} \frac{16\sqrt{2}C}{\sqrt{1-\ln 2}} \right) \\
 &\leq 32 \ln(1/\mu) + S \left(\frac{1+4C}{2} + 2^{(S-1)/2} \frac{16\sqrt{2}C}{\sqrt{1-\ln 2}} \right) \\
 &\leq 32 \ln(1/\mu) + S \left(\frac{1+4C}{2} + \frac{\sqrt{2(1-\ln 2)}}{\sqrt{\delta'} \cdot \sqrt{2}} \frac{16\sqrt{2}C}{\sqrt{1-\ln 2}} \right) \\
 &\leq 32 \ln(1/\mu) + \log_2(2(1-\ln 2)/\delta') \left(\frac{1+4C}{2} + \frac{32C}{\sqrt{2\delta'}} \right) \\
 &\leq 32 \ln(1/\mu) + \log_2(2/\varepsilon) \left(\frac{1}{2} + 2C + \frac{32C}{\sqrt{\varepsilon}} \right) \\
 &\leq \frac{32 \ln(1/\mu) + \log_2(2/\varepsilon) (1+34C)}{\sqrt{\varepsilon}},
 \end{aligned}$$

where we used that $2^S < 2(1-\ln 2)/\delta'$ by definition of S (noting that $S \geq 1$), as well as $\delta' = \varepsilon/2$ and $\varepsilon \leq 1$. \square

Corollary 14.2.6. *Let $\mu > 0$, let $\mathbf{A} \in [\mu, 1]^{n \times n}$ be a rational matrix with $\|\mathbf{A}\|_1 \leq 1$, let $\mathbf{r}, \mathbf{c} \in (0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, and let $\varepsilon \in (0, 1]$. Then there exists a quantum algorithm that, given sparse query access to \mathbf{A} , with probability $\geq 2/3$, computes $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n$ such that $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is $\varepsilon\text{-}\ell^1$ -scaled to (\mathbf{r}, \mathbf{c}) , at a total time complexity of $\tilde{O}(n^{1.5}/\varepsilon^2)$.*

14.3. Randomized quantum Sinkhorn algorithm

In this section we discuss a randomized version of the Sinkhorn algorithm, and sketch its analysis. Classically, this randomized version is of interest due to its good performance in practice (asymptotically the complexity is not better than the usual Sinkhorn algorithm). It is therefore natural to ask if it admits a similar quantum speedup. In this section we give an affirmative answer.

The algorithm is stated in Algorithm 14.2.¹ It differs from the ordinary Sinkhorn algorithm in that, rather than updating all rows or all columns in each iteration, it selects a random row or column in each iteration, and only updates this row or column. Because rows and columns have m/n non-zero entries on average, this results in an *expected* quantum time complexity of $\tilde{O}(\sqrt{m/(n\varepsilon)})$ per iteration, where

¹Note that Algorithm 14.2 returns $\mathbf{x}^{(\tau-1)}$ and $\mathbf{y}^{(\tau-1)}$ and could therefore stop after τ iterations. However, for the sake of simplifying the analysis, we always continue for T iterations.

$\varepsilon > 0$ is the desired precision (measured in relative entropy). Furthermore, we show that $\tilde{O}(n/\varepsilon)$ iterations suffice to obtain an ε - ℓ^1 -scaling with probability $\geq 2/3$. Note that we do not necessarily alternate between choosing rows or columns. Suitable choices of parameters for Algorithm 14.2 are given in Theorem 14.3.4, along with the corresponding guarantees and the resulting quantum time complexity.

The analysis of the randomized Sinkhorn algorithm is somewhat more involved because we are no longer able to test whether the matrix is ε -scaled at every iteration. Indeed, this test has a quantum time complexity of roughly \sqrt{mn}/ε and would therefore lead to a complexity of roughly $\sqrt{mn}/\varepsilon \cdot n/\varepsilon$, which is worse than guaranteed by the classical Sinkhorn algorithm! Similarly, naively maintaining all marginals in a data structure is prohibitively expensive. Therefore, while the algorithm is running, we do not know whether the potential is still decreasing every iteration, and may lose progress during subsequent iterations. However, we can show that for any probability $p > 0$ and subsequent appropriate choices of parameters (in particular for large enough T), a $1 - p$ fraction of the produced iterates $(x^{(t)}, y^{(t)})$ yield an ε -relative-entropy-scaling. This implies that a uniformly random choice of stopping iteration yields an ε -relative-entropy-scaling with probability $1 - p$.

Algorithm 14.2: Randomized Sinkhorn with finite precision and failure probability

Input: Query access to $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and non-zero entries at least $\mu > 0$, target marginals $\mathbf{r}, \mathbf{c} \in [0, 1]^n$ with $\|\mathbf{r}\|_1 = \|\mathbf{c}\|_1 = 1$, iteration count $T \geq 0$, bit counts $b_1, b_2 \geq 0$, precision $\delta \in (0, 1)$ and subroutine failure probability $\eta \in [0, 1]$

Output: Vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with entries encoded in (b_1, b_2) fixed-point format

Analysis: Theorem 14.3.4

```

1  $\mathbf{x}^{(0)}, \mathbf{y}^{(0)} \leftarrow \mathbf{0}$ ;
2 for  $t \leftarrow 1, 2, \dots, T$  do
3     Pick  $\beta \in \{0, 1\}$  uniformly at random;
4     Pick  $\ell \in [n]$  uniformly at random;
5     if  $\beta = 0$  then
6          $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)}$ ;  $\mathbf{y}^{(t)} \leftarrow \mathbf{y}^{(t-1)}$ ;
7          $x_\ell^{(t)} \leftarrow \text{LogSumExp}(\mathbf{A}_{\ell \bullet}, r_\ell, \mathbf{y}^{(t-1)}, \delta, b_1, b_2, \eta, \mu)$ ;
8     else
9          $\mathbf{y}^{(t)} \leftarrow \mathbf{y}^{(t-1)}$ ;  $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)}$ ;
10         $y_\ell^{(t)} \leftarrow \text{LogSumExp}(\mathbf{A}_{\bullet \ell}, c_\ell, \mathbf{x}^{(t-1)}, \delta, b_1, b_2, \eta, \mu)$ ;
11    end if
12 end for
13 Pick  $\tau \in [T]$  uniformly at random;
14 return  $(\mathbf{x}^{(\tau-1)}, \mathbf{y}^{(\tau-1)})$ ;
```

The following lemma bounds the progress of a row or column update (if we ignore the effects of the truncation) in terms of the quantity $\rho(a||b) = b - a + a \ln \frac{a}{b}$

14. Quantum Sinkhorn and Osborne algorithms

defined in Section 13.2.2.

Lemma 14.3.1. *Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, let $\delta \in [0, 1]$ and $\ell \in [n]$, and let $\hat{\mathbf{x}} \in \mathbb{R}^n$ be a vector such that $|\hat{x}_\ell - \ln(r_\ell / \sum_{j=1}^n A_{\ell j} e^{y_j})| \leq \delta$, and for every $k \neq \ell$, $\hat{x}_k = x_k$. Then*

$$f(\mathbf{x}, \mathbf{y}) - f(\hat{\mathbf{x}}, \mathbf{y}) \geq \rho(r_\ell \| r_\ell(\mathbf{A}(\mathbf{x}, \mathbf{y}))) - 2\delta r_\ell.$$

A similar statement holds for an update of y_ℓ (using c_ℓ rather than r_ℓ).

The lemma plays a similar role as Lemma 14.1.2 and its proof is completely analogous. We can use the lemma to lower bound the *expected* progress in each iteration in terms of the relative entropy $D(\mathbf{a} \| \mathbf{b}) = \sum_{\ell=1}^n \rho(a_\ell \| b_\ell)$, since the latter is directly related to the expectation of $\rho(a_\ell \| b_\ell)$ for uniformly random $\ell \in [n]$. To formally analyze the expected progress it will be useful to define the following events.

Definition 14.3.2. For $t = 1, \dots, T$, we define the following events:

- Let G_t denote the event that $D(\mathbf{r} \| \mathbf{r}(\mathbf{A}^{(t-1)})) \leq \varepsilon$ and $D(\mathbf{c} \| \mathbf{c}(\mathbf{A}^{(t-1)})) \leq \varepsilon$.
- Let S_t denote the event that the call to `LogSumExp` on line 7 succeeds (if $\beta = 0$ in this iteration) or that the call to `LogSumExp` on line 10 succeeds (if $\beta = 1$ in this iteration).
- Define S to be the intersection of the events S_t , i.e., $S = \bigcap_{t=1}^T S_t$.

Let us also abbreviate $f_t = f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})$. Then we have the following key lemma which bounds the expected progress in terms of the probability of \overline{G}_t .

Lemma 14.3.3. *Assume $\Pr[S_t] \geq 1 - \eta$ for $t \in [T]$. Then, for any $t \in [T]$, we have*

$$\mathbb{E}[(f_{t-1} - f_t) \mathbb{1}_S] \geq \frac{\varepsilon}{2n} \Pr[\overline{G}_t] - \varepsilon \eta T - \frac{2\delta}{n}.$$

Proof. In this proof we use that for independent random variables X, Y and any function $h(x, y)$ we have

$$\mathbb{E}[h(X, Y)] = \sum_{y \in Y} \Pr[Y = y] \mathbb{E}[h(X, y)]. \quad (14.3.1)$$

For $t \in [T]$, let $\beta^{(t)}$ be the random choice of row versus column scaling made on Line 3, and let $\ell^{(t)} \in [n]$ be the random index chosen on Line 4. Then, Lemma 14.3.1 shows that if S holds then we have

$$f_{t-1} - f_t \geq \rho(M_t \| m_t) - 2\delta M_t, \text{ where } (M_t, m_t) = \begin{cases} (r_{\ell^{(t)}}, r_{\ell^{(t)}}(\mathbf{A}^{(t-1)})) & \text{if } \beta^{(t)} = 0, \\ (c_{\ell^{(t)}}, c_{\ell^{(t)}}(\mathbf{A}^{(t-1)})) & \text{if } \beta^{(t)} = 1. \end{cases} \quad (14.3.2)$$

We define $R_t = \min\{\rho(M_t \| m_t), \varepsilon\} - 2\delta M_t$. Then the above implies

$$\mathbb{E}[(f_{t-1} - f_t) \mathbb{1}_S] \geq \mathbb{E}[R_t \mathbb{1}_S].$$

We may expand this lower bound as

$$\mathbb{E}[R_t \mathbb{1}_S] = \mathbb{E}[R_t] - \mathbb{E}[R_t \mathbb{1}_{\bar{S}}] = \mathbb{E}[R_t \mathbb{1}_{\overline{G_t}}] + \mathbb{E}[R_t \mathbb{1}_{G_t}] - \mathbb{E}[R_t \mathbb{1}_{\bar{S}}]. \quad (14.3.3)$$

To lower bound the first term, recall that $(\beta^{(t)}, \ell^{(t)})$ are drawn independently from $\mathbf{A}^{(t-1)}$, while the event G_t only depends on $\mathbf{A}^{(t-1)}$ (and hence is independent from $(\beta^{(t)}, \ell^{(t)})$). Therefore, using Eq. (14.3.1) we can lower bound

$$\begin{aligned} & \mathbb{E}[R_t \mathbb{1}_{\overline{G_t}}] \\ &= \mathbb{E}[(\min\{\rho(M_t \| m_t), \varepsilon\} - 2\delta M_t) \mathbb{1}_{\overline{G_t}}] \\ &= \frac{1}{2n} \mathbb{E} \left[\sum_{\ell=1}^n (\min\{\rho(r_\ell \| r_\ell(\mathbf{A}^{(t-1)})), \varepsilon\} + \min\{\rho(c_\ell \| c_\ell(\mathbf{A}^{(t-1)})), \varepsilon\} - 2\delta(r_\ell + c_\ell)) \mathbb{1}_{\overline{G_t}} \right] \\ &= \frac{1}{2n} \mathbb{E} \left[\sum_{\ell=1}^n (\min\{\rho(r_\ell \| r_\ell(\mathbf{A}^{(t-1)})), \varepsilon\} + \min\{\rho(c_\ell \| c_\ell(\mathbf{A}^{(t-1)})), \varepsilon\}) \mathbb{1}_{\overline{G_t}} \right] - \frac{1}{2n} \mathbb{E}[4\delta \mathbb{1}_{\overline{G_t}}] \\ &\geq \frac{1}{2n} \mathbb{E} \left[\min \left\{ \sum_{\ell=1}^n \rho(r_\ell \| r_\ell(\mathbf{A}^{(t-1)})) + \rho(c_\ell \| c_\ell(\mathbf{A}^{(t-1)})), \varepsilon \right\} \mathbb{1}_{\overline{G_t}} \right] - \frac{2\delta}{n} \Pr[\overline{G_t}] \\ &= \frac{1}{2n} \mathbb{E} \left[\min \left\{ D(\mathbf{r} \| \mathbf{r}(\mathbf{A}^{(t-1)})) + D(\mathbf{c} \| \mathbf{c}(\mathbf{A}^{(t-1)})), \varepsilon \right\} \mathbb{1}_{\overline{G_t}} \right] - \frac{2\delta}{n} \Pr[\overline{G_t}] \\ &= \frac{1}{2n} \mathbb{E} \left[\varepsilon \mathbb{1}_{\overline{G_t}} \right] - \frac{2\delta}{n} \Pr[\overline{G_t}] = \left(\frac{\varepsilon}{2n} - \frac{2\delta}{n} \right) \Pr[\overline{G_t}] \end{aligned}$$

where we first used the inequality $\sum_{\ell=1}^n \min\{a_\ell, b\} \geq \min\{\sum_{\ell=1}^n a_\ell, b\}$, which holds for any real numbers $a_1, \dots, a_n \in \mathbb{R}$ and $b \geq 0$, and we then used that $D(\mathbf{r} \| \mathbf{r}(\mathbf{A}^{(t-1)})) + D(\mathbf{c} \| \mathbf{c}(\mathbf{A}^{(t-1)})) \geq \varepsilon$ whenever G_t does not hold.

To lower bound $\mathbb{E}[R_t \mathbb{1}_{G_t}]$, note that we also have the bound $R_t \geq -2\delta M_t$ as $\min\{\rho(M_t \| m_t), \varepsilon\}$ is non-negative, so again using independence of $(\beta^{(t)}, \ell^{(t)})$ from G_t , we obtain

$$\mathbb{E}[R_t \mathbb{1}_{G_t}] \geq -\frac{2\delta}{n} \Pr[G_t] = \frac{2\delta}{n} \Pr[\overline{G_t}] - \frac{2\delta}{n}.$$

Lastly, to upper bound $\mathbb{E}[R_t \mathbb{1}_{\bar{S}}]$, note that $R_t \leq \varepsilon$, so

$$\mathbb{E}[R_t \mathbb{1}_{\bar{S}}] \leq \varepsilon \Pr[\bar{S}] \leq \varepsilon \eta T$$

where the last step follows from the union bound. Combining the bounds in Eq. (14.3.3) then yields the desired bound. \square

Lemma 14.1.4 shows how large we need to take b_1 to ensure that all components of $(\mathbf{x}^{(t)}, \mathbf{y}^{(t)})$ are in the interval $[-2^{b_1}, 2^{b_1}]$.²

Finally, we can combine Theorem 14.1.1 and Lemma 14.3.3 to show that Algorithm 14.2 returns an ε -relative-entropy-scaling (with probability $\geq 1 - p$). This is also an $O(\sqrt{\varepsilon})$ - ℓ^1 -scaling by Pinsker's inequality.

²Technically, Lemma 14.1.4 is about updating all entries of either \mathbf{x} or \mathbf{y} in each iteration, however its proof shows that the same bound also applies if we update only a single coordinate per iteration.

14. Quantum Sinkhorn and Osborne algorithms

Theorem 14.3.4. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a matrix whose non-zero entries are at least $\mu > 0$ and $\|\mathbf{A}\|_1 \leq 1$. Assume that \mathbf{A} is asymptotically scalable to (\mathbf{r}, \mathbf{c}) . Let $p \in (0, 1]$ and $\varepsilon \in (0, 1]$. Choose*

$$T = \left\lceil \frac{6n}{\varepsilon p} \ln\left(\frac{1}{\mu}\right) \right\rceil,$$

$\delta = \frac{\varepsilon p}{12}$, $\eta = \frac{p}{6nT}$, $b_2 = \lceil \log_2(1/\delta) \rceil$, and $b_1 = \lceil \log_2(T) + \log_2(\ln(\frac{1}{\mu}) + 1 + \sigma) \rceil$, where $\sigma = \max(|\ln r_{\min}|, |\ln c_{\min}|)$. Then, Algorithm 14.2 with these parameters and given sparse query access to \mathbf{A} , with probability $\geq 1 - p$, returns a pair (\mathbf{x}, \mathbf{y}) such that $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is ε -relative-entropy-scaled to (\mathbf{r}, \mathbf{c}) , for an expected quantum time complexity of $\tilde{O}(\sqrt{mn}/(\varepsilon p)^{1.5})$.

Before we prove Theorem 14.3.4 we make the following two small remarks about its formulation. First, for a constant success probability (say 2/3), the time complexity “on expectation” can be converted to a worst-case time complexity using Markov’s inequality. Second, while the number of iterations T scales inverse-polynomially with the failure probability p , one can obtain a logarithmic scaling with the failure probability in the following way: take $O(\ln(1/p))$ many independent runs of Algorithm 14.2 with $p = 1/3$ and use TestScaling on the outputs. With probability $1 - p$ one of the outputs will provide an ε -relative-entropy scaling.

Proof of Theorem 14.3.4. By Theorem 14.1.1, we have

$$\mathbb{E}[(f_0 - f_T)\mathbb{1}_S] \leq \mathbb{E}[\ln(\frac{1}{\mu})\mathbb{1}_S] = \ln(\frac{1}{\mu}) \Pr[S] \leq \ln(\frac{1}{\mu}). \quad (14.3.4)$$

using $\mu \leq 1$. We now lower bound the left-hand side by a telescoping sum, using Lemma 14.3.3,

$$\begin{aligned} \mathbb{E}[(f_0 - f_T)\mathbb{1}_S] &= \sum_{t=1}^T \mathbb{E}[(f_{t-1} - f_t)\mathbb{1}_S] \\ &\geq \sum_{t=1}^T \left(\frac{\varepsilon}{2n} \Pr[\overline{G}_t] - \varepsilon \eta T - \frac{2\delta}{n} \right) \\ &= T \left(\frac{\varepsilon}{2n} - \varepsilon \eta T - \frac{2\delta}{n} \right) - \frac{\varepsilon}{2n} \sum_{t=1}^T \Pr[G_t]. \end{aligned}$$

Together with Eq. (14.3.4), we obtain for uniformly random $\tau \in [T]$ the bound

$$\begin{aligned} \Pr[G_\tau] &= \frac{1}{T} \sum_{t=1}^T \Pr[G_t] \geq 1 - 2n\eta T - \frac{4\delta}{\varepsilon} - \frac{2n}{\varepsilon} \frac{\ln(\frac{1}{\mu})}{T} \\ &= 1 - \left(2n \cdot \frac{p}{6n(\frac{6n}{\varepsilon p} \ln(\frac{1}{\mu}))} \cdot \frac{6n}{\varepsilon p} \ln(\frac{1}{\mu}) \right) - \left(\frac{4\varepsilon p}{12\varepsilon} \right) - \left(\frac{2n}{\varepsilon} \frac{\ln(\frac{1}{\mu})}{\frac{6n}{\varepsilon p} \ln(\frac{1}{\mu})} \right) = 1 - p. \end{aligned} \quad (14.3.5)$$

Since we return $(\mathbf{x}^{(\tau-1)}, \mathbf{y}^{(\tau-1)})$ for $\tau \in [T]$ independently and uniformly at random, $\Pr[G_\tau]$ is the success probability of our algorithm and the first equality in Eq. (14.3.5) holds. This shows the output of Algorithm 14.2 satisfies the guarantees of the theorem.

Finally, the time complexity of Algorithm 14.2 follows from Theorem 13.3.2 and an application of Jensen's inequality (using concavity of the square root function). Indeed, the complexity of applying `LogSumExp` to a row or column with s possibly non-zero entries is $\tilde{O}(\sqrt{s/\delta})$ quantumly, and hence the expected time complexity per application of `LogSumExp` is $\tilde{O}(\sqrt{m/(n\delta)})$ quantumly. The total expected time complexity then follows from linearity of expectation and the bound on the number of iterations T . \square

14.4. Randomized quantum Osborne algorithm

In this section we present an algorithm for the matrix-balancing problem (Problem 13.2.7). The algorithm that we analyze is a quantum version of Osborne's algorithm with random updates [Osb60]; the latter was very recently analyzed in the classical setting by Altschuler and Parrilo [AP23].

The goal of balancing a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ is to find a vector $\mathbf{x} \in \mathbb{R}^n$ such that the matrix $\mathbf{A}(\mathbf{x}) = \mathbf{A}(\mathbf{x}, -\mathbf{x})$ has (approximately) equal row and column sums. As explained in Section 13.2.3, we assume without loss of generality that the diagonal entries of \mathbf{A} are zero; and we also assume that every row and every column of \mathbf{A} contains at least one non-zero entry.

Osborne's algorithm is an alternating iterative algorithm for matrix balancing that proceeds similarly to Sinkhorn's algorithm for matrix scaling. The idea is to focus on an individual coordinate at a time. That is, given an index $\ell \in [n]$, we would like to update \mathbf{x} to $\mathbf{x}' = \mathbf{x} + \Delta_\ell \mathbf{e}_\ell$, where Δ_ℓ is chosen such that

$$r_\ell(\mathbf{A}(\mathbf{x}')) = c_\ell(\mathbf{A}(\mathbf{x}')).$$

Expanding the above equation and using $A_{\ell\ell} = 0$ yields

$$e^{\Delta_\ell} \cdot r_\ell(\mathbf{A}(\mathbf{x})) = e^{-\Delta_\ell} \cdot c_\ell(\mathbf{A}(\mathbf{x})).$$

Since we assume every row and column contains at least one non-zero entry, the above equation has a unique solution, given by

$$\Delta_\ell = \ln \left(\sqrt{\frac{c_\ell(\mathbf{A}(\mathbf{x}))}{r_\ell(\mathbf{A}(\mathbf{x}))}} \right). \quad (14.4.1)$$

Note that the updates of multiple coordinates *cannot* be done simultaneously, since each x_ℓ can potentially affect *all* row and column marginals at the same time. This is in contrast with the Sinkhorn algorithm for matrix scaling, where all row scalings or all column scalings can be updated at the same time. Therefore a choice must be made as to which index to update at each iteration. Altschuler and Parrilo [AP23] analyze several such choices, including greedy, random and cyclic variants. The greedy and random variants have better guaranteed performance than the cyclic variant. However, to implement the greedy version, one has to maintain an auxiliary data structure which contains all current row and column marginals, which can be updated after each iteration with a cost of order $O(n)$. Therefore, even though we can accelerate each iteration with quantum approximate counting to become sublinear in n , the greedy iterations would still incur at least a

cost of order n for the updates. So instead, we focus on a randomized variant of Osborne's algorithm.

The analysis in the quantum setting is more complicated than in the classical case. The basic argument follows similar lines to the one given in [AP23] (a potential argument like for matrix scaling, but with the relative entropy replaced by the Hellinger distance). However, in the classical setting, one can increase the precision of individual updates at a very small cost, and one does not have to deal with the possibility of making backwards progress. In the quantum setting, in contrast, we do not have this luxury: we cannot test whether the matrix is ε - H^2 -balanced each iteration, and the relatively high imprecision of the updates can cause subsequent iterations to destroy this property. This situation is similar to the one discussed in Section 14.3 for the randomized Sinkhorn algorithm, and we adapt the ideas developed in that section in the analysis here.

Algorithm 14.3: Random Osborne with finite precision and failure probability

Input: Query access to $\mathbf{A} \in [0, 1]^{n \times n}$ and non-zero entries at least $\mu > 0$, iteration count $T \geq 0$, bit counts $b_1, b_2 \geq 0$, update precision $\delta \in (0, 1)$ and subroutine failure probability $\eta \in [0, 1]$

Output: Vector $\mathbf{x} \in \mathbb{R}^n$ with entries encoded in (b_1, b_2) fixed-point format

Analysis: Theorem 14.4.6

```

1  $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$ ; // entries in  $(b_1, b_2)$  fixed-point format
2 for  $t \leftarrow 1, 2, \dots, T$  do
3   Pick  $\ell \in [n]$  uniformly at random;
4    $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)}$ ;
5    $x_\ell^{(t)} \leftarrow \frac{1}{2} \text{LogSumExp}(\mathbf{A}_{\ell \bullet}, 1, -\mathbf{x}^{(t-1)}, \delta, b_1, b_2, \eta/2, \mu)$ –
6    $\frac{1}{2} \text{LogSumExp}(\mathbf{A}_{\bullet \ell}, 1, \mathbf{x}^{(t-1)}, \delta, b_1, b_2, \eta/2, \mu)$ ;
7 end for
8 Pick  $\tau \in [T]$  uniformly at random;
9 return  $(\mathbf{x}^{(\tau)}, \mathbf{y}^{(\tau)})$ ;
```

Our randomized version of Osborne's algorithm is given in Algorithm 14.3. It allows for an additive error δ in computing the update as compared to Eq. (14.4.1). To see that for $\delta = 0$ the update in Algorithm 14.3 is exactly the same as in Eq. (14.4.1), one can rewrite

$$\begin{aligned}
x_\ell + \Delta_\ell &= x_\ell + \ln \left(\sqrt{\frac{c_\ell(\mathbf{A}(\mathbf{x}))}{r_\ell(\mathbf{A}(\mathbf{x}))}} \right) \\
&= x_\ell + \ln \left(\sqrt{\frac{\sum_{i=1}^n A_{i\ell} e^{x_i - x_\ell}}{\sum_{j=1}^n A_{\ell j} e^{x_\ell - x_j}}} \right) \\
&= \frac{1}{2} \left(\ln \left(\frac{1}{\sum_{j=1}^n A_{\ell j} e^{-x_j}} \right) - \ln \left(\frac{1}{\sum_{i=1}^n A_{i\ell} e^{x_i}} \right) \right).
\end{aligned}$$

In each iteration of Algorithm 14.3, the two calls to the LogSumExp subroutine compute the two logarithms to additive precision δ . Hence each iteration computes

an approximation to the ideal Osborne update with additive precision δ (assuming no errors).

To analyze Algorithm 14.3, we consider the convex potential $f: \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = \sum_{i,j=1}^n A_{ij} e^{x_i - x_j} = \|\mathbf{A}(\mathbf{x})\|_1, \quad (14.4.2)$$

in analogy with the potential for the analysis of matrix scaling. Let f^* be the infimum of $f(\mathbf{x})$.

We first state a lower bound on the potential.

Lemma 14.4.1. *If $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ is asymptotically balanceable and its non-zero entries are at least $\mu > 0$, then $f^* \geq \mu$.*

This can be deduced immediately from Proposition 5.1.2. The next lemma then gives a lower bound on the progress made by an approximate Osborne update.

Lemma 14.4.2. *Let $\ell \in [n]$ be an index, $\mathbf{x} \in \mathbb{R}^n$ be a vector, $\delta \in [0, 1]$, and let \mathbf{x}' be the vector with $x'_k = x_k$ for $k \neq \ell$ and*

$$\left| x'_\ell - \left(x_\ell + \ln \left(\sqrt{\frac{c_\ell(\mathbf{A}(\mathbf{x}))}{r_\ell(\mathbf{A}(\mathbf{x}))}} \right) \right) \right| \leq \delta, \quad (14.4.3)$$

i.e., \mathbf{x}' is a δ -additive approximation of the Osborne update of \mathbf{x} for the ℓ -th index. Then

$$f(\mathbf{x}) - f(\mathbf{x}') \geq \left(\sqrt{r_\ell(\mathbf{A}(\mathbf{x}))} - \sqrt{c_\ell(\mathbf{A}(\mathbf{x}))} \right)^2 - 2\delta \sqrt{r_\ell(\mathbf{A}(\mathbf{x}))c_\ell(\mathbf{A}(\mathbf{x}))}.$$

Proof. Note that $\mathbf{A}(\mathbf{x}')$ and $\mathbf{A}(\mathbf{x})$ have the same entries outside of the ℓ -th row and column. Expanding the definition and recalling that $A_{\ell\ell} = 0$ gives

$$f(\mathbf{x}) - f(\mathbf{x}') = r_\ell(\mathbf{A}(\mathbf{x})) + c_\ell(\mathbf{A}(\mathbf{x})) - r_\ell(\mathbf{A}(\mathbf{x}')) - c_\ell(\mathbf{A}(\mathbf{x}')).$$

For convenience, write $z = x'_\ell - x_\ell - \ln(\sqrt{c_\ell(\mathbf{A}(\mathbf{x}))/r_\ell(\mathbf{A}(\mathbf{x}))}) \in [-\delta, \delta]$. Then

$$\begin{aligned} r_\ell(\mathbf{A}(\mathbf{x}')) &= e^{x'_\ell - x_\ell} \cdot r_\ell(\mathbf{A}(\mathbf{x})) = e^z \cdot \sqrt{r_\ell(\mathbf{A}(\mathbf{x}))c_\ell(\mathbf{A}(\mathbf{x}))}, \\ c_\ell(\mathbf{A}(\mathbf{x}')) &= e^{x_\ell - x'_\ell} \cdot c_\ell(\mathbf{A}(\mathbf{x})) = e^{-z} \cdot \sqrt{r_\ell(\mathbf{A}(\mathbf{x}))c_\ell(\mathbf{A}(\mathbf{x}))}. \end{aligned}$$

Since $|z| \leq |\delta| \leq 1$, we have the estimate $e^z + e^{-z} \leq 2 + 2|z| \leq 2 + 2\delta$, which yields

$$f(\mathbf{x}) - f(\mathbf{x}') \geq r_\ell(\mathbf{A}(\mathbf{x})) + c_\ell(\mathbf{A}(\mathbf{x})) - (2 + 2\delta) \sqrt{r_\ell(\mathbf{A}(\mathbf{x}))c_\ell(\mathbf{A}(\mathbf{x}))}$$

as desired. \square

As a corollary, we obtain the following relation between approximate minimizers of f and balancings:

Corollary 14.4.3. *Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ be asymptotically balanceable, let $\xi > 0$, and assume $\mathbf{x} \in \mathbb{R}^n$ is such that $f(\mathbf{x}) - f^* \leq \xi f(\mathbf{x})$. Then $\mathbf{A}(\mathbf{x})$ is $n\xi$ - H^2 -balanced, and $2\sqrt{\xi}$ - ℓ^2 -balanced.*

14. Quantum Sinkhorn and Osborne algorithms

Proof. Averaging over the choice of $\ell \in [n]$ in Lemma 14.4.2 yields

$$\xi \|\mathbf{A}(\mathbf{x})\|_1 \geq f(\mathbf{x}) - f^* \geq \frac{1}{n} \sum_{\ell=1}^n \left(\sqrt{r_\ell(\mathbf{A}(\mathbf{x}))} - \sqrt{c_\ell(\mathbf{A}(\mathbf{x}))} \right)^2 = \frac{1}{n} H^2(\mathbf{r}(\mathbf{A}(\mathbf{x})), \mathbf{c}(\mathbf{A}(\mathbf{x}))).$$

This shows that $\mathbf{A}(\mathbf{x})$ is $n\xi$ - H^2 -balanced. Similarly, taking the average with respect to the weights $w_\ell = (\sqrt{r_\ell(\mathbf{A}(\mathbf{x}))} + \sqrt{c_\ell(\mathbf{A}(\mathbf{x}))})^2 \leq 2(r_\ell(\mathbf{A}(\mathbf{x})) + c_\ell(\mathbf{A}(\mathbf{x})))$ yields

$$\begin{aligned} \xi \|\mathbf{A}(\mathbf{x})\|_1 \geq f(\mathbf{x}) - f^* &\geq \frac{\sum_{\ell=1}^n w_\ell \left(\sqrt{r_\ell(\mathbf{A}(\mathbf{x}))} - \sqrt{c_\ell(\mathbf{A}(\mathbf{x}))} \right)^2}{\sum_{\ell=1}^n w_\ell} \\ &= \frac{\sum_{\ell=1}^n (r_\ell(\mathbf{A}(\mathbf{x})) - c_\ell(\mathbf{A}(\mathbf{x})))^2}{\sum_{\ell=1}^n w_\ell} \\ &\geq \frac{\|\mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x}))\|_2^2}{4\|\mathbf{A}(\mathbf{x})\|_1}, \end{aligned}$$

showing that $\mathbf{A}(\mathbf{x})$ is $2\sqrt{\xi}$ - ℓ^2 -balanced. \square

Remark 14.4.4. One can show that $\mathbf{A}(\mathbf{x})$ is $2\sqrt{\xi}$ - ℓ^2 -balanced when $\ln f(\mathbf{x}) - \ln f^* \leq \xi$, which is a weaker assumption (since $\ln z \geq 1 - \frac{1}{z}$ for $z = f(\mathbf{x})/f^* \geq 1$). This may be deduced from the fact that $\ln f(\mathbf{x})$ is a 2-smooth convex function with gradient $(\mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x}))) / \|\mathbf{A}(\mathbf{x})\|_1$, see the proof of Theorem 14.1.1.

The following lemma gives a suitable choice for the parameter b_1 which ensures correct functioning of the algorithm, i.e., ensures that the requirements of LogSumExp are always satisfied.

Lemma 14.4.5. Let $\mathbf{x}^{(0)} = \mathbf{0}$ and $T \in \mathbb{N}$. Suppose that for all $t \in [T]$, $\mathbf{x}^{(t)}$ is a δ -additive approximation of an Osborne update for $\mathbf{x}^{(t-1)}$ as in Eq. (14.4.3). Then we have the following bound for all $t \leq T$,

$$\|\mathbf{x}^{(t)}\|_\infty \leq t \cdot \left(\frac{1}{2} \ln(\|\mathbf{A}\|_1/\mu) + \delta \right). \quad (14.4.4)$$

Then choosing $b_1 = \lceil \log_2(\max_\ell \{|\ln(r_\ell(\mathbf{A}))|, |\ln(c_\ell(\mathbf{A}))|\} + T \cdot (\ln(\|\mathbf{A}\|_1/\mu)/2 + 1)) \rceil$, guarantees that $2^{b_1} \geq |\ln(\sum_{j=1}^n A_{\ell j} e^{-x_j^{(t)}})|$ and $2^{b_1} \geq |\ln(\sum_{i=1}^n A_{i\ell} e^{x_i^{(t)}})|$ for any $\ell \in [n]$ and $t \leq T$.

Proof. We show that if \mathbf{x} is any vector, then any vector \mathbf{x}' obtained by a δ -additive approximate Osborne update of \mathbf{x} satisfies

$$\|\mathbf{x}'\|_\infty \leq \|\mathbf{x}\|_\infty + \frac{1}{2} \ln(\|\mathbf{A}\|_1/\mu) + \delta. \quad (14.4.5)$$

Suppose \mathbf{x}' is obtained by updating the ℓ -th index of \mathbf{x} , i.e.,

$$\left| x'_\ell - \left(x_\ell + \ln \sqrt{\frac{c_\ell(\mathbf{A}(\mathbf{x}))}{r_\ell(\mathbf{A}(\mathbf{x}))}} \right) \right| \leq \delta.$$

Then observe that

$$x_\ell + \ln \sqrt{\frac{c_\ell(\mathbf{A}(\mathbf{x}))}{r_\ell(\mathbf{A}(\mathbf{x}))}} = x_\ell + \ln \sqrt{\frac{\sum_{i=1}^n A_{i\ell} e^{x_i - x_\ell}}{\sum_{j=1}^n A_{\ell j} e^{x_\ell - x_j}}} = \ln \sqrt{\frac{\sum_{i=1}^n A_{i\ell} e^{x_i}}{\sum_{j=1}^n A_{\ell j} e^{-x_j}}}.$$

Since we have

$$\sqrt{\frac{\mu}{\|\mathbf{A}\|_1}} \cdot e^{-\|\mathbf{x}\|_\infty} \leq \sqrt{\frac{\sum_{i=1}^n A_{i\ell} e^{x_i}}{\sum_{j=1}^n A_{\ell j} e^{-x_j}}} \leq \sqrt{\frac{\|\mathbf{A}\|_1}{\mu}} \cdot e^{\|\mathbf{x}\|_\infty},$$

the updated coordinate x'_ℓ satisfies

$$|x'_\ell| \leq \|\mathbf{x}\|_\infty + \frac{1}{2} \ln(\|\mathbf{A}\|_1/\mu) + \delta.$$

Since all other coordinates of \mathbf{x}' and \mathbf{x} agree and $\|\mathbf{A}\|_1 \geq \mu$, the same upper bound holds for $\|\mathbf{x}'\|_\infty$. Thus we have proved Eq. (14.4.5), and Eq. (14.4.4) now follows by induction.

As a consequence of Eq. (14.4.4), for every $t \leq T$ and $\ell \in [n]$, we have that

$$\left| \ln \left(\sum_{j=1}^n A_{\ell j} e^{-x_j^{(t)}} \right) - \ln(r_\ell(\mathbf{A})) \right| \leq \|\mathbf{x}^{(t)}\|_\infty \leq t \cdot \left(\frac{1}{2} \ln(\|\mathbf{A}\|_1/\mu) + \delta \right),$$

$$\left| \ln \left(\sum_{i=1}^n A_{i\ell} e^{x_i^{(t)}} \right) - \ln(c_\ell(\mathbf{A})) \right| \leq \|\mathbf{x}^{(t)}\|_\infty \leq t \cdot \left(\frac{1}{2} \ln(\|\mathbf{A}\|_1/\mu) + \delta \right).$$

This implies the second statement. \square

Following a similar proof strategy as in Section 14.3, we show that Algorithm 14.3 finds approximate balancings in a certain number of iterations. For the time complexity we use the quantum implementation of the LogSumExp subroutine from Section 13.3.

Theorem 14.4.6. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with diagonal entries zero, each row and column containing at least one non-zero element and all non-zero entries at least $\mu > 0$. Assume \mathbf{A} is asymptotically balanceable, and let $\varepsilon \in (0, 1]$ and $p \in (0, 1]$. Choose*

$$T = \left\lceil \frac{3n \ln(\|\mathbf{A}\|_1/\mu)}{p\varepsilon} \right\rceil,$$

as well as $\delta = p\varepsilon/6$, $\eta = p\varepsilon/(3nT)$, $b_1 = \lceil \log_2(\sigma + T \cdot (\ln(\|\mathbf{A}\|_1/\mu)/2 + 1)) \rceil$, where $\sigma = \max_\ell \{|\ln(r_\ell(\mathbf{A}))|, |\ln(c_\ell(\mathbf{A}))|\}$, and $b_2 = \lceil \log_2(1/\delta) \rceil$. Then Algorithm 14.3 with these parameters returns a vector \mathbf{x} such that $\mathbf{A}(\mathbf{x})$ is ε - H^2 -balanced with probability at least $1 - p$. The time complexity is $\tilde{O}(\sqrt{mn}/(p^{1.5}\varepsilon^{1.5}))$ on expectation where m is the number of non-zero entries of \mathbf{A} .

Proof. We proceed in analogy with the analysis of randomized Sinkhorn. Let S_t denote the event that both calls to LogSumExp in the t -th iteration of Algorithm 14.3 succeed, and let S be the intersection of all these events for $t \in [T]$. Then $\Pr[\overline{S}_t] \leq \eta$, since two calls are made and each call fails with probability at most $\eta/2$, and also

14. Quantum Sinkhorn and Osborne algorithms

$\Pr[\bar{S}] \leq \eta T$ by the union bound. Let G_t be the event that $\mathbf{A}(\mathbf{x}^{(t-1)})$ is ε - H^2 -balanced, and let $\ell^{(t)}$ be the choice of index in the t -th iteration. For convenience, we abbreviate $F(\mathbf{x}) = \ln f(\mathbf{x})$, $F_t = \ln f_t = \ln f(\mathbf{x}^{(t)})$, $\mathbf{A}^{(t)} = \mathbf{A}(\mathbf{x}^{(t)})$, $\phi_t = r_{\ell^{(t)}}(\mathbf{A}^{(t-1)})$, and $\psi_t = c_{\ell^{(t)}}(\mathbf{A}^{(t-1)})$. Lemma 14.4.2 implies that if S holds, then

$$f_{t-1} - f_t \geq \left(\sqrt{\phi_t} - \sqrt{\psi_t} \right)^2 - 2\delta \sqrt{\phi_t \psi_t}.$$

Dividing by f_{t-1} and rearranging yields

$$\frac{f_t}{f_{t-1}} \leq 1 - \frac{1}{f_{t-1}} \left(\sqrt{\phi_t} - \sqrt{\psi_t} \right)^2 + \frac{2\delta}{f_{t-1}} \sqrt{\phi_t \psi_t}.$$

The quantity on the right-hand side is positive since $f_{t-1} \geq \phi_t + \psi_t > (\sqrt{\phi_t} - \sqrt{\psi_t})^2$ (the first inequality uses that \mathbf{A} has zero diagonal) and ϕ_t, ψ_t are both positive, so taking logarithms and using the estimate $\ln(1+z) \leq z$ gives

$$F_t - F_{t-1} \leq \frac{1}{f_{t-1}} \left(2\delta \sqrt{\phi_t \psi_t} - \left(\sqrt{\phi_t} - \sqrt{\psi_t} \right)^2 \right).$$

Define a random variable R_t by

$$R_t = \frac{1}{f_{t-1}} \left(\left(\sqrt{\phi_t} - \sqrt{\psi_t} \right)^2 - 2\delta \sqrt{\phi_t \psi_t} \right).$$

Then the above estimates yield

$$\mathbb{E}[(F_{t-1} - F_t) \mathbb{1}_S] \geq \mathbb{E}[R_t \mathbb{1}_S]. \quad (14.4.6)$$

We now expand the right-hand side as

$$\mathbb{E}[R_t \mathbb{1}_S] = \mathbb{E}[R_t \mathbb{1}_{G_t}] + \mathbb{E}[R_t \mathbb{1}_{\bar{G}_t}] - \mathbb{E}[R_t \mathbb{1}_{\bar{S}}], \quad (14.4.7)$$

and bound the terms individually. The first term in Eq. (14.4.7) can be bounded as

$$\begin{aligned} \mathbb{E}[R_t \mathbb{1}_{G_t}] &\geq \mathbb{E} \left[-\frac{2\delta}{f_{t-1}} \sqrt{\phi_t \psi_t} \mathbb{1}_{G_t} \right] \\ &\geq -\mathbb{E} \left[\frac{\delta}{f_{t-1}} (\phi_t + \psi_t) \mathbb{1}_{G_t} \right] \\ &= -\delta \mathbb{E} \left[\frac{r_{\ell^{(t)}}(\mathbf{A}^{(t-1)}) + c_{\ell^{(t)}}(\mathbf{A}^{(t-1)})}{\|\mathbf{A}^{(t-1)}\|_1} \mathbb{1}_{G_t} \right] \\ &= -\delta \mathbb{E} \left[\frac{1}{n} \sum_{\ell=1}^n \frac{r_{\ell}(\mathbf{A}^{(t-1)}) + c_{\ell}(\mathbf{A}^{(t-1)})}{\|\mathbf{A}^{(t-1)}\|_1} \mathbb{1}_{G_t} \right] \\ &= -\frac{2\delta}{n} \Pr[G_t], \end{aligned}$$

where the first inequality is obtained by discarding a positive term, the second follows from the arithmetic-geometric mean inequality, and the second to last equality

follows from the independence of $\ell^{(t)}$ and $\mathbf{A}^{(t-1)}$, which allows us to first average over $\ell^{(t)}$ and then over $\mathbf{A}^{(t-1)}$, using that $\sum_{\ell=1}^n r_{\ell}(\mathbf{A}^{(t-1)}) = \sum_{\ell=1}^n c_{\ell}(\mathbf{A}^{(t-1)}) = \|\mathbf{A}^{(t-1)}\|_1$.

For the second term of Eq. (14.4.7), we bound

$$\begin{aligned}
 \mathbb{E}[\mathbf{R}_t \mathbb{1}_{\overline{\mathbf{G}}_t}] &\geq \mathbb{E}\left[\frac{1}{f_{t-1}} \left(\sqrt{\phi_t} - \sqrt{\psi_t}\right)^2 \mathbb{1}_{\overline{\mathbf{G}}_t}\right] - \frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t] \\
 &= \mathbb{E}\left[\frac{\left(\sqrt{r_{\ell^{(t)}}(\mathbf{A}^{(t-1)})} - \sqrt{c_{\ell^{(t)}}(\mathbf{A}^{(t-1)})}\right)^2}{\|\mathbf{A}^{(t-1)}\|_1} \mathbb{1}_{\overline{\mathbf{G}}_t}\right] - \frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t] \\
 &= \mathbb{E}\left[\frac{1}{n} \sum_{\ell=1}^n \frac{\left(\sqrt{r_{\ell}(\mathbf{A}^{(t-1)})} - \sqrt{c_{\ell}(\mathbf{A}^{(t-1)})}\right)^2}{\|\mathbf{A}^{(t-1)}\|_1} \mathbb{1}_{\overline{\mathbf{G}}_t}\right] - \frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t] \\
 &= \mathbb{E}\left[\frac{1}{n} \frac{\left\|\sqrt{\mathbf{r}(\mathbf{A}^{(t-1)})} - \sqrt{\mathbf{c}(\mathbf{A}^{(t-1)})}\right\|_2^2}{\|\mathbf{A}^{(t-1)}\|_1} \mathbb{1}_{\overline{\mathbf{G}}_t}\right] - \frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t] \\
 &\geq \frac{\varepsilon}{n} \Pr[\overline{\mathbf{G}}_t] - \frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t],
 \end{aligned}$$

where the first inequality is derived as in the lower bound for $\mathbb{E}[\mathbf{R}_t \mathbb{1}_{\mathbf{G}_t}]$, the second equality holds by independence of $\ell^{(t)}$ from $\mathbf{A}^{(t-1)}$, the second inequality is Lemma 13.2.4 (the lower bound on the Hellinger distance), and the last inequality holds since $\mathbf{A}^{(t-1)}$ is not ε - H^2 -balanced in the event $\overline{\mathbf{G}}_t$.

Lastly, to lower bound $\mathbb{E}[\mathbf{R}_t \mathbb{1}_{\overline{\mathbf{S}}}]$ in Eq. (14.4.7), note that

$$\mathbf{R}_t \leq \frac{(\sqrt{\phi_t} - \sqrt{\psi_t})^2}{f_{t-1}} \leq \frac{\phi_t + \psi_t}{f_{t-1}} = \frac{r_{\ell^{(t)}}(\mathbf{A}^{(t-1)}) + c_{\ell^{(t)}}(\mathbf{A}^{(t-1)})}{\|\mathbf{A}^{(t-1)}\|_1} \leq 1,$$

where in the last inequality we use that $A_{\ell^{(t)}\ell^{(t)}} = 0$. Therefore, we obtain

$$\mathbb{E}[\mathbf{R}_t \mathbb{1}_{\overline{\mathbf{S}}}] \leq \Pr[\overline{\mathbf{S}}] \leq \eta T.$$

Using Eq. (14.4.6) and the three bounds just derived, we have

$$\begin{aligned}
 \mathbb{E}[(F_{t-1} - F_t) \mathbb{1}_{\mathbf{S}}] &\geq \mathbb{E}[\mathbf{R}_t \mathbb{1}_{\mathbf{S}}] = \mathbb{E}[\mathbf{R}_t \mathbb{1}_{\mathbf{G}_t}] + \mathbb{E}[\mathbf{R}_t \mathbb{1}_{\overline{\mathbf{G}}_t}] - \mathbb{E}[\mathbf{R}_t \mathbb{1}_{\overline{\mathbf{S}}}] \\
 &\geq -\frac{2\delta}{n} \Pr[\overline{\mathbf{G}}_t] + \frac{\varepsilon}{n} \Pr[\overline{\mathbf{G}}_t] - \frac{2\delta}{n} \Pr[\mathbf{G}_t] - \eta T \\
 &= \frac{\varepsilon}{n} \Pr[\overline{\mathbf{G}}_t] - \frac{2\delta}{n} - \eta T.
 \end{aligned}$$

Since we also have $F_0 - F_T \leq \ln(\|\mathbf{A}\|_1) - \ln(\mu) = \ln(\|\mathbf{A}\|_1/\mu)$ by Lemma 14.4.1, we can finish up the argument in exactly the same manner as for Theorem 14.3.4. Indeed, we have a telescoping sum

$$\ln(\|\mathbf{A}\|_1/\mu) \geq \mathbb{E}[(F_0 - F_T) \mathbb{1}_{\mathbf{S}}] = \sum_{t=1}^T \mathbb{E}[(F_{t-1} - F_t) \mathbb{1}_{\mathbf{S}}]$$

14. Quantum Sinkhorn and Osborne algorithms

$$\geq \frac{\varepsilon}{n} \sum_{t=1}^T \Pr[\overline{G}_t] - T \left(\frac{2\delta}{n} + \eta T \right).$$

In other words, we have

$$\sum_{t=1}^T \Pr[\overline{G}_t] \leq \frac{1}{\varepsilon} \left(n \ln(\|\mathbf{A}\|_1/\mu) + 2\delta T + \eta n T^2 \right).$$

Since the choice of stopping time $\tau \in [T]$ is uniformly random, we have

$$\begin{aligned} \Pr[\overline{G}_\tau] &= \frac{1}{T} \sum_{t=1}^T \Pr[\overline{G}_t] \leq \frac{1}{T\varepsilon} \left(n \ln(\|\mathbf{A}\|_1/\mu) + 2\delta T + \eta n T^2 \right) \\ &= \frac{n \ln(\|\mathbf{A}\|_1/\mu)}{T\varepsilon} + \frac{2\delta}{\varepsilon} + \frac{\eta n T}{\varepsilon} \leq p, \end{aligned}$$

where the last inequality follows from the choice of parameters. In other words, when we take the stopping time $\tau \in [T]$ uniformly random, the probability that $\mathbf{x}^{(\tau-1)}$ provides an ε - H^2 -balancing is at least $1 - p$.

The claimed time complexity follows by multiplying the number of iterations T with the cost of `LogSumExp` with $\delta = p\varepsilon/6$, which is $\tilde{O}(\sqrt{n/\delta} \log(1/\eta))$ as detailed in Theorem 13.3.2. \square

As in the matrix-scaling setting, one can convert the “on expectation” time complexity to a worst-case guarantee, and improve the dependence on the failure probability from inverse polynomial to logarithmic by repeating and testing whether the output yields a balanced matrix. (One can implement a procedure `TestBalancing` similar to `TestScaling` to test whether $\mathbf{A}(\mathbf{x})$ is ε - H^2 -balanced, with success probability $1 - \eta$, for a quantum cost of $\tilde{O}(\sqrt{mn}/\varepsilon \log(1/\eta))$, see Proposition 13.3.4.) By using Lemma 13.2.4 to convert the guarantee in squared Hellinger distance to ℓ^1 , we obtain the following corollary.

Corollary 14.4.7. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with all zeroes on the diagonal, each row and column containing at least one non-zero element and all non-zero entries at least $\mu > 0$. Assume \mathbf{A} is asymptotically balanceable, and let $\varepsilon \in (0, 1]$ and $p \in (0, 1]$. Then there exists a quantum algorithm that, given sparse query access to \mathbf{A} , returns a vector \mathbf{x} such that $\mathbf{A}(\mathbf{x})$ is ε - ℓ^1 -balanced with probability at least $1 - p$. The time complexity is $\tilde{O}(\sqrt{mn}/(p^{1.5}\varepsilon^3))$ on expectation where m is the number of non-zero entries of \mathbf{A} .*

The Osborne and Sinkhorn algorithms are special cases of a more general algorithm for a more general problem (see, e.g., [CMTV17; BLNW20] for details and motivations). Suppose one is given a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ and a vector $\mathbf{d} \in \mathbb{R}^n$, and one wishes to find \mathbf{x} such that

$$r_\ell(\mathbf{A}(\mathbf{x})) - c_\ell(\mathbf{A}(\mathbf{x})) = d_\ell$$

for each $\ell \in [n]$. That is, one prescribes the differences between the row and column sums. One can again solve this for individual $\ell \in [n]$, by expanding the above equation and solving for x_ℓ . It is clear that the case $\mathbf{d} = \mathbf{0}$ amounts to the matrix balancing problem and the procedure just described is the Osborne algorithm. On

the other hand, the matrix scaling problem for $\mathbf{B} \in \mathbb{R}_{\geq 0}^{n \times n}$ and targets $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{> 0}^n$ can be modeled by the above problem for the choices

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{B} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{d} = (\mathbf{r}, -\mathbf{c}) \in \mathbb{R}^{2n},$$

so that the first n constraints yield the desired constraints on the row marginals, and the last n yield the desired constraints on the column marginals. Note that the support of this matrix \mathbf{A} is such that we may simultaneously update the first n coordinates (or the last n coordinates) since their updates are independent. This leads to the Sinkhorn algorithm. More generally, if G is the directed graph with vertex set $[n]$ and adjacency defined by the support of \mathbf{A} , then any subset of vertices which form an independent set in G can be updated simultaneously (cf. [AP23, Sec. 2.5 & App. B]). In general, one can give an explicit expression for the updates, and analyze the progress via the potential $\mathbf{x} \mapsto \|\mathbf{A}(\mathbf{x})\|_1 - \langle \mathbf{d}, \mathbf{x} \rangle$. This potential generalizes the ones we used for matrix scaling and matrix balancing (up to a change of sign for the last n variables in the former case), and it admits similar potential bounds and lower bounds on the progress as derived above. However, the details of carrying out such an analysis are less clear. We do note here that the second-order methods in Chapter 15 should work as they do for matrix scaling and balancing (in particular we expect little difficulty for *classical* algorithms), but leave a detailed analysis to future work.

15. Quantum box-constrained Newton methods

In this chapter, we show how to obtain a quantum speed-up based on the box-constrained Newton method for matrix scaling and balancing from [CMTV17], with the main result being Theorems 15.2.8 and 15.3.7, and its consequences for matrix scaling given in Corollaries 15.2.9 and 15.2.10 and for balancing given in Corollary 15.3.8. The time complexity of both algorithms depends on a diameter bound R_∞ for approximate solutions. This diameter bound can be a constant for some special cases, e.g. for entrywise-positive matrices in matrix scaling. This allows us to achieve better quantum speed-ups for such instances as compared to the first-order methods of Chapter 14.

In Section 15.1, we first recall some of the concepts that are used in the algorithm, including the definition of second-order robust convex functions, the notion of a k -oracle, and a theorem regarding efficient (classical) implementation of a k -oracle for the class of symmetric diagonally-dominant matrices with non-positive off-diagonal entries. We then show that for a second-order robust function $g: \mathbb{R}^n \rightarrow \mathbb{R}$ and a given $\mathbf{x} \in \mathbb{R}^n$ such that the sublevel set $\{\mathbf{x}' : g(\mathbf{x}') \leq g(\mathbf{x})\}$ is bounded, one can use a k -oracle and approximations to the gradient and Hessian of g to find a vector \mathbf{x}' such that the potential gap $g(\mathbf{x}') - g(\mathbf{x}^*)$ is smaller than $g(\mathbf{x}) - g(\mathbf{x}^*)$ where \mathbf{x}^* is a minimizer of g . This result extends [CMTV17, Thm. 3.4] to a setting where one can only obtain rough approximations of the gradient and Hessian of g . Next, in Section 15.2 and Section 15.3 respectively, we then show that this strategy applies to regularized versions \tilde{f} of the potential f for matrix scaling (Eq. (13.1.3)) and balancing (Eq. (14.4.2)). The Hessian of the potential function is closely related to graph Laplacians and symmetric diagonally-dominant matrices, which allows us to apply the recent quantum algorithm for graph sparsification (Theorem 13.3.5) to approximate the Hessian of \tilde{f} . As for the gradients, we can approximate them in ℓ^1 -norm using Corollary 13.3.1 (after accounting for the rescaling or rebalancing determined by \mathbf{x}).

15.1. Minimizing second-order robust convex functions

In what follows we will minimize a convex function (potential) that satisfies a certain regularity condition: its Hessian can be approximated well on an ℓ^∞ -norm ball.

Definition 15.1.1 ([CMTV17, Def. 3.1]). A convex function $g: \mathbb{R}^n \rightarrow \mathbb{R}$ is called *second-order robust* with respect to ℓ^∞ if for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with $\|\mathbf{x} - \mathbf{y}\|_\infty \leq 1$,

$$\frac{1}{e^2} \text{Hess } g(\mathbf{x}) \leq \text{Hess } g(\mathbf{y}) \leq e^2 \text{Hess } g(\mathbf{x}).$$

This chapter is adapted from [GN22].

15. Quantum box-constrained Newton methods

This implies that the local quadratic approximation to g has good quality on a small ℓ^∞ -norm ball. It is therefore natural to consider the problem of minimizing a convex quadratic function over an ℓ^∞ -norm ball. We will use the following notion.

Definition 15.1.2 (k -oracle). An algorithm \mathcal{A} is called a k -oracle for a class of matrices $\mathcal{M} \subseteq \mathbb{R}^{n \times n}$ if for input (\mathbf{H}, \mathbf{b}) with $\mathbf{H} \in \mathcal{M}$, $\mathbf{b} \in \mathbb{R}^n$, it returns a vector $\mathbf{x} \in \mathbb{R}^n$ such that $\|\mathbf{x}\|_\infty \leq k$ and

$$\frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle \leq \frac{1}{2} \cdot \min_{\|z\|_\infty \leq 1} \left(\frac{1}{2}z^\top \mathbf{H}z + \langle \mathbf{b}, z \rangle \right). \quad (15.1.1)$$

Definition 15.1.3 (SDD matrix). A matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is called *symmetric diagonally-dominant* if it is symmetric, and for every $i \in [n]$, one has $A_{ii} \geq \sum_{j \neq i} |A_{ij}|$.

In [CMTV17] it is shown how to efficiently implement an $O(\log(n))$ -oracle for any SDD matrix \mathbf{H} whose off-diagonal entries are non-positive. Their algorithm uses an efficient construction of a *vertex sparsifier chain* of \mathbf{H} due to [LPS15; KLP+16].

Theorem 15.1.4 ([CMTV17, Thm. 5.11]). *Given a classical description of an SDD matrix $\mathbf{H} \in \mathbb{R}^{n \times n}$ with $\tilde{O}(m)$ non-zero entries, such that $H_{ij} \leq 0$ for $i \neq j$, and a classical vector $\mathbf{b} \in \mathbb{R}^n$, we can find in time $\tilde{O}(m)$ a vector $\mathbf{x} \in \mathbb{R}^n$ such that $\|\mathbf{x}\|_\infty = O(\log n)$ and*

$$\frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle \leq \frac{1}{2} \cdot \min_{\|z\|_\infty \leq 1} \left(\frac{1}{2}z^\top \mathbf{H}z + \langle \mathbf{b}, z \rangle \right).$$

A k -oracle \mathcal{A} gives rise to an iterative method for minimizing a second-order robust function g : starting from $\mathbf{x}_0 \in \mathbb{R}^n$, we define a sequence $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ by

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \frac{1}{k}\Delta_i, \quad \Delta_i = \mathcal{A} \left(\frac{e^2}{k^2}\mathbf{H}_i, \frac{1}{k}\mathbf{b}_i \right)$$

where \mathbf{H}_i is an approximate Hessian at $\mathbf{x}^{(i)}$, and \mathbf{b}_i is an approximate gradient at $\mathbf{x}^{(i)}$. The following theorem, which is an adaptation of [CMTV17, Thm. 3.4], lower bounds the progress made in each iteration.

Theorem 15.1.5. *Let $g: \mathbb{R}^n \rightarrow \mathbb{R}$ be a second-order robust function with respect to ℓ^∞ , let $\mathbf{x} \in \mathbb{R}^n$ be a starting point, and suppose \mathbf{x}^* is a minimizer of g . Assume that we are given*

(1) a vector $\mathbf{b} \in \mathbb{R}^n$ such that

$$\|\mathbf{b} - \text{grad } g(\mathbf{x})\|_1 \leq \delta,$$

(2) two SDD matrices \mathbf{H}_m and \mathbf{H}_a with non-positive off-diagonal entries, such that there exists $\delta_a \geq 0$ and symmetric \mathbf{H}'_m and \mathbf{H}'_a satisfying $\text{Hess } g(\mathbf{x}) = \mathbf{H}'_m + \mathbf{H}'_a$ and

$$\frac{2}{3}\mathbf{H}_m \leq \mathbf{H}'_m \leq \frac{4}{3}\mathbf{H}_m, \quad \|\mathbf{H}_a - \mathbf{H}'_a\|_1 \leq \delta_a.$$

Let $k = O(\log n)$ be such that there exists a k -oracle \mathcal{A} for the class of SDD-matrices with non-positive off-diagonal entries (Theorem 15.1.4). Then for $\mathbf{H} = \mathbf{H}_m + \mathbf{H}_a$ and $\Delta = \mathcal{A}\left(\frac{4e^2}{3k^2}\mathbf{H}, \frac{1}{k}\mathbf{b}\right)$, the vector $\mathbf{x}' = \mathbf{x} + \frac{1}{k}\Delta$ satisfies

$$g(\mathbf{x}') - g(\mathbf{x}^*) \leq \left(1 - \frac{1}{4e^4 \max(kR_\infty, 1)}\right) (g(\mathbf{x}) - g(\mathbf{x}^*)) + e^2\delta_a + \frac{3}{2}\delta,$$

where R_∞ is the ℓ^∞ -radius of the sublevel set $\{\mathbf{x}' : g(\mathbf{x}') \leq g(\mathbf{x})\}$ about \mathbf{x} .

Before giving the proof, we introduce the following notation. For a symmetric matrix \mathbf{H} and $\mathbf{b}, \mathbf{z} \in \mathbb{R}^n$, we denote

$$Q(\mathbf{H}, \mathbf{b}, \mathbf{z}) = \langle \mathbf{b}, \mathbf{z} \rangle + \frac{1}{2}\mathbf{z}^\top \mathbf{H} \mathbf{z}.$$

We will use the following easily-verified properties of Q repeatedly.

Lemma 15.1.6. *For symmetric matrices \mathbf{H}, \mathbf{H}' and vectors $\mathbf{b}, \mathbf{b}', \mathbf{z}$, we have the following estimates:*

(i) *If $\mathbf{H} \leq \mathbf{H}'$, then $Q(\mathbf{H}, \mathbf{b}, \mathbf{z}) \leq Q(\mathbf{H}', \mathbf{b}, \mathbf{z})$.*

(ii) *If $\|\mathbf{H} - \mathbf{H}'\|_1 \leq \delta_a$, then*

$$|Q(\mathbf{H}, \mathbf{b}, \mathbf{z}) - Q(\mathbf{H}', \mathbf{b}, \mathbf{z})| \leq \frac{1}{2}\delta_a \|\mathbf{z}\|_\infty^2.$$

(iii) *We have*

$$|Q(\mathbf{H}, \mathbf{b}, \mathbf{z}) - Q(\mathbf{H}, \mathbf{b}', \mathbf{z})| = |\langle \mathbf{b} - \mathbf{b}', \mathbf{z} \rangle| \leq \|\mathbf{b} - \mathbf{b}'\|_1 \|\mathbf{z}\|_\infty.$$

Proof of Theorem 15.1.5. We follow the proof of [CMTV17, Thm. 3.4], and use their implementation of a k -oracle \mathcal{A} for $k = O(\log n)$, as detailed in Theorem 15.1.4. That is, \mathcal{A} takes as input an SDD matrix \mathbf{H} with $\tilde{O}(m)$ non-zero entries (off-diagonal entries ≤ 0) and a vector \mathbf{b} , and outputs a vector \mathbf{z} such that $\|\mathbf{z}\|_\infty \leq k$ and

$$Q(\mathbf{H}, \mathbf{b}, \mathbf{z}) \leq \frac{1}{2} \inf_{\|\mathbf{z}'\|_\infty \leq 1} Q(\mathbf{H}, \mathbf{b}, \mathbf{z}').$$

Then for

$$\mathbf{x}' = \mathbf{x} + \frac{1}{k}\Delta, \quad \Delta = \mathcal{A}\left(\frac{4e^2}{3k^2}\mathbf{H}, \frac{1}{k}\mathbf{b}\right)$$

we have

$$\begin{aligned} Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \frac{1}{k}\Delta\right) &= Q\left(\frac{4e^2}{3k^2}\mathbf{H}, \frac{1}{k}\mathbf{b}, \Delta\right) \\ &\leq \frac{1}{2} \inf_{\|\mathbf{v}\|_\infty \leq 1} Q\left(\frac{4e^2}{3k^2}\mathbf{H}, \frac{1}{k}\mathbf{b}, \mathbf{v}\right) \\ &= \frac{1}{2} \inf_{\|\mathbf{v}\|_\infty \leq 1} Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \mathbf{v}/k\right) \\ &= \frac{1}{2} \inf_{\|\mathbf{v}\|_\infty \leq \frac{1}{k}} Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \mathbf{v}\right). \end{aligned} \tag{15.1.2}$$

Note that the second-order robustness of g implies that for $\tilde{\mathbf{x}} \in \mathbb{R}^n$ with $\|\mathbf{x} - \tilde{\mathbf{x}}\|_\infty \leq 1$, we have quadratic lower and upper bounds

$$Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \tilde{\mathbf{x}} - \mathbf{x}\right) \leq g(\tilde{\mathbf{x}}) - g(\mathbf{x}) \leq Q\left(e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \tilde{\mathbf{x}} - \mathbf{x}\right). \quad (15.1.3)$$

The remainder of the proof is structured as follows. We first compare quadratics involving $\text{Hess } g(\mathbf{x})$ and $\nabla g(\mathbf{x})$ to quadratics involving the approximations \mathbf{H} and \mathbf{b} in Eqs. (15.1.4) and (15.1.5). Using these estimates we then obtain a local progress bound over an ℓ^∞ -ball of radius $1/k$, see Eq. (15.1.6). Finally, we convert this local bound into a more global estimate.

Using Lemma 15.1.6, the properties of the approximate Hessian and gradient guarantee that

$$\begin{aligned} & Q\left(e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \tilde{\mathbf{x}} - \mathbf{x}\right) \\ & \leq Q\left(e^2 \text{Hess } g(\mathbf{x}), \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + \delta \\ & = Q\left(e^2 \mathbf{H}'_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(e^2 \mathbf{H}'_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle + \delta \\ & \leq Q\left(\frac{4e^2}{3} \mathbf{H}_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(e^2 \mathbf{H}_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + \frac{e^2}{2} \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle + \delta \\ & \leq Q\left(\frac{4e^2}{3} \mathbf{H}_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(\frac{4e^2}{3} \mathbf{H}_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + \frac{e^2}{2} \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle + \delta \\ & = Q\left(\frac{4e^2}{3} \mathbf{H}, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + \frac{e^2}{2} \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 + \delta. \end{aligned} \quad (15.1.4)$$

Furthermore, we also have the upper bound on the quadratic term:

$$\begin{aligned} & Q\left(\frac{4e^2}{3} \mathbf{H}, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) \\ & = Q\left(\frac{4e^2}{3} \mathbf{H}_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(\frac{4e^2}{3} \mathbf{H}_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle \\ & \leq Q\left(2e^2 \mathbf{H}'_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(2e^2 \mathbf{H}_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle \\ & \leq Q\left(2e^2 \mathbf{H}'_m, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + Q\left(2e^2 \mathbf{H}'_a, \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + e^2 \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 - \langle \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x} \rangle \\ & = Q\left(2e^2 \text{Hess } g(\mathbf{x}), \mathbf{b}, \tilde{\mathbf{x}} - \mathbf{x}\right) + e^2 \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 \\ & \leq Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \tilde{\mathbf{x}} - \mathbf{x}\right) + e^2 \delta_a \|\tilde{\mathbf{x}} - \mathbf{x}\|_\infty^2 + \delta. \end{aligned} \quad (15.1.5)$$

Let \mathbf{v}_L and \mathbf{v}_U be the minimizers of quadratics over the ℓ^∞ -ball of radius $1/k$:

$$\begin{aligned} \mathbf{v}_L &= \underset{\|\mathbf{v}\|_\infty \leq 1/k}{\text{argmin}} Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}\right), \\ \mathbf{v}_U &= \underset{\|\mathbf{v}\|_\infty \leq 1/k}{\text{argmin}} Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}\right). \end{aligned}$$

We can further bound Eq. (15.1.2):

$$\begin{aligned} Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \frac{1}{k}\Delta\right) &\leq \frac{1}{2} \inf_{\|\mathbf{v}\|_\infty \leq 1/k} Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \mathbf{v}\right) \\ &\leq \frac{1}{2} \inf_{\|\mathbf{v}\|_\infty \leq 1/k} \left(Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}\right) + e^2\delta_a \|\mathbf{v}\|_\infty^2 + \delta\right) \\ &\leq \frac{1}{2} Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_U\right) + \frac{e^2\delta_a}{2k^2} + \frac{1}{2}\delta, \end{aligned}$$

where the second inequality uses Eq. (15.1.5), and the norm bounds $\|\mathbf{v}\|_\infty \leq 1/k \leq 1$ (to apply the inequality). Using the quadratic upper bound from Eq. (15.1.3) on $g(\mathbf{x} + \frac{1}{k}\Delta) - g(\mathbf{x})$ and Eq. (15.1.4), this yields

$$\begin{aligned} g(\mathbf{x} + \frac{1}{k}\Delta) - g(\mathbf{x}) &\leq Q\left(e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \frac{1}{k}\Delta\right) \\ &\leq Q\left(\frac{4e^2}{3}\mathbf{H}, \mathbf{b}, \frac{1}{k}\Delta\right) + \frac{e^2}{2}\delta_a \|\frac{1}{k}\Delta\|_\infty^2 + \delta \\ &\leq \frac{1}{2} Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_U\right) + e^2\delta_a + \frac{3}{2}\delta, \end{aligned}$$

where the last inequality uses that $\|\frac{1}{k}\Delta\|_\infty^2 \leq 1$ and $\frac{e^2\delta_a}{2k^2} + \frac{e^2\delta_a}{2} \leq e^2\delta_a$. We can then further upper bound the quadratic term using

$$\begin{aligned} Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_U\right) &\leq Q\left(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \frac{\mathbf{v}_L}{2e^4}\right) \\ &= \frac{1}{2e^4} Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_L\right), \end{aligned}$$

where the inequality uses $\mathbf{v}_U = \operatorname{argmin}_{\|\mathbf{v}\|_\infty \leq 1/k} Q(2e^2 \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v})$ and $\|\mathbf{v}_L\|_\infty \leq 1/k$. Collecting estimates, we obtain

$$g(\mathbf{x} + \frac{1}{k}\Delta) - g(\mathbf{x}) \leq \frac{1}{4e^4} Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_L\right) + e^2\delta_a + \frac{3}{2}\delta. \quad (15.1.6)$$

We now convert this to a more global estimate. Let \mathbf{x}^* be a global minimizer of g . Set $\mathbf{y} = \mathbf{x} + \frac{1}{\max(kR_\infty, 1)}(\mathbf{x}^* - \mathbf{x})$, so that $\|\mathbf{y} - \mathbf{x}\|_\infty \leq \frac{1}{k}$. For the lower bound

$$g_L(\tilde{\mathbf{x}}) = g(\mathbf{x}) + Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \tilde{\mathbf{x}} - \mathbf{x}\right)$$

on $g(\tilde{\mathbf{x}})$ we see that $g_L(\mathbf{x} + \mathbf{v}_L) \leq g_L(\mathbf{y}) \leq g(\mathbf{y})$ since $\mathbf{x} + \mathbf{v}_L$ minimizes $g_L \leq g$ over the ℓ^∞ -ball of radius $1/k$ around \mathbf{x} . By convexity of g we get

$$g(\mathbf{y}) = g\left(\mathbf{x} + \frac{1}{\max(kR_\infty, 1)}(\mathbf{x}^* - \mathbf{x})\right) \leq \left(1 - \frac{1}{\max(kR_\infty, 1)}\right)g(\mathbf{x}) + \frac{1}{\max(kR_\infty, 1)}g(\mathbf{x}^*)$$

so

$$Q\left(\frac{1}{e^2} \text{Hess } g(\mathbf{x}), \text{grad } g(\mathbf{x}), \mathbf{v}_L\right) = g_L(\mathbf{x} + \mathbf{v}_L) - g(\mathbf{x})$$

$$\leq g(\mathbf{y}) - g(\mathbf{x}) \leq \frac{1}{\max(kR_\infty, 1)}(g(\mathbf{x}^*) - g(\mathbf{x})).$$

Using the estimate in Eq. (15.1.6), this gives

$$g(\mathbf{x}) - g(\mathbf{x} + \frac{1}{k}\Delta) \geq \frac{1}{4e^4 \max(kR_\infty, 1)}(g(\mathbf{x}) - g(\mathbf{x}^*)) - (e^2\delta_\alpha + \frac{3}{2}\delta), \quad (15.1.7)$$

which after rearranging and rewriting $\mathbf{x}' = \mathbf{x} + \frac{1}{k}\Delta$ reads

$$g(\mathbf{x}') - g(\mathbf{x}^*) \leq \left(1 - \frac{1}{4e^4 \max(kR_\infty, 1)}\right)(g(\mathbf{x}) - g(\mathbf{x}^*)) + e^2\delta_\alpha + \frac{3}{2}\delta. \quad \square$$

15.2. Quantum box-constrained matrix scaling

We show how to combine the box-constrained Newton method and the quantum Laplacian sparsifier from Theorem 13.3.5 to solve matrix scaling. For this, we sequentially present: (1) a regularized potential function for matrix scaling, (2) quantum algorithms for approximating its Hessian and gradient, (3) a quantum box-constrained matrix scaling algorithm and its analysis. We note that our algorithm does not just iteratively apply the result of Section 15.1: inbetween such iterations we must change our scaling vectors to ensure the 1-norm of the rescaling $\mathbf{A}(\mathbf{x}, \mathbf{y})$ does not become too large.

15.2.1. A second-order robust potential function for matrix scaling

For a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$, a desired error $\varepsilon > 0$, and some number $B \geq 1$, we consider the regularized potential function $\tilde{f}(\mathbf{x}, \mathbf{y})$ given by

$$\tilde{f}(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}, \mathbf{y}) + \frac{\varepsilon}{ne^B} \left(\sum_i (e^{x_i} + e^{-x_i}) + \sum_j (e^{y_j} + e^{-y_j}) \right), \quad (15.2.1)$$

where $f(\mathbf{x}, \mathbf{y}) = \sum_{i,j=1}^n A_{ij} e^{x_i + y_j} - \langle \mathbf{r}, \mathbf{x} \rangle - \langle \mathbf{c}, \mathbf{y} \rangle$ is the commonly-used potential function from Eq. (13.1.3). The regularization term is taken from [CMTV17], but with a different weight (since we aim for scalings with respect to relative entropy rather than with respect to ℓ^2 -distance). The following is then an adaptation of [CMTV17, Lem. 4.10].

Lemma 15.2.1. *Assume \mathbf{A} is asymptotically scalable, with $\|\mathbf{A}\|_1 \leq 1$, and $\mu > 0$ its smallest non-zero entry. Let $B > 0$ and $\varepsilon > 0$ be given. Then the regularized potential \tilde{f} satisfies the following properties:*

- (i) \tilde{f} is second-order robust with respect to ℓ^∞ , and its Hessian is SDD;
- (ii) we have $f(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{x}, \mathbf{y})$ for any (\mathbf{x}, \mathbf{y}) ;
- (iii) for all (\mathbf{x}, \mathbf{y}) such that $\tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0}, \mathbf{0})$, we have $\|(\mathbf{x}, \mathbf{y})\|_\infty \leq B + \ln(4n + (n \ln(1/\mu)/\varepsilon))$, and
- (iv) for any $(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)$ such that $f(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) \leq f^* + \varepsilon$ and $\|(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)\|_\infty \leq B$, one has $\tilde{f}(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) \leq f^* + 5\varepsilon$. In particular, if such a $(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)$ exists, then $|f^* - \tilde{f}^*| \leq 5\varepsilon$.

Proof. The first point is easy to verify, as is the second point (the regularization term is always positive). For the third point, suppose we have a (\mathbf{x}, \mathbf{y}) such that $\tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0})$. Then

$$\frac{\varepsilon}{ne^B} \left(\sum_i (e^{x_i} + e^{-x_i}) + \sum_j (e^{y_j} + e^{-y_j}) \right) \leq f(\mathbf{0}, \mathbf{0}) - f(\mathbf{x}, \mathbf{y}) + \frac{\varepsilon}{ne^B} \cdot 4n \leq \ln(1/\mu) + \frac{4\varepsilon}{e^B}, \quad (15.2.2)$$

where the last inequality follows from the potential bound $f(\mathbf{0}, \mathbf{0}) - f^* \leq \ln(1/\mu)$ (Theorem 14.1.1), which depends on $\|\mathbf{A}\|_1 \leq 1$; in general the upper bound is $\|\mathbf{A}\|_1 - 1 + \ln(1/\mu)$. Since each of the regularization terms is positive, we may restrict ourselves to a single term and see that

$$e^{x_i} + e^{-x_i} \leq \frac{ne^B \ln(1/\mu)}{\varepsilon} + 4n,$$

from which we may deduce

$$|x_i| \leq \ln \left(\frac{e^B n \ln(1/\mu)}{\varepsilon} + 4n \right) = B + \ln \left(\frac{n \ln(1/\mu)}{\varepsilon} + \frac{4n}{e^B} \right) \leq B + \ln \left(\frac{n \ln(1/\mu)}{\varepsilon} + 4n \right),$$

where the last inequality uses $e^B \geq 1$ (recall $B > 0$). The same upper bound holds for $|y_j|$.

For the last point, note that if $\mathbf{x}_\varepsilon = (x_1, \dots, x_n)$, then $e^{x_i} + e^{-x_i} \leq 2e^B$ and similarly for y_i , so

$$\tilde{f}(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) \leq f(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) + \frac{\varepsilon}{ne^B} \cdot 4ne^B = f(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) + 4\varepsilon \leq f^* + 5\varepsilon.$$

If such a \mathbf{z}_ε exists, then

$$f^* \leq \tilde{f}^* \leq \tilde{f}(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) \leq f^* + 5\varepsilon. \quad \square$$

15.2.2. Quantum algorithms for approximating the Hessian and gradient for scaling

In order to use Theorem 15.1.5 to minimize \tilde{f} , we need to show how to approximate both the gradient and Hessian of \tilde{f} . We first consider the Hessian of \tilde{f} , which can be written as the sum of the Hessian of f and the Hessian of the regularizer $\tilde{f} - f$. We have

$$\begin{aligned} \text{Hess } f(\mathbf{x}, \mathbf{y}) &= \begin{bmatrix} \text{diag}(\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) & \mathbf{A}(\mathbf{x}, \mathbf{y}) \\ \mathbf{A}(\mathbf{x}, \mathbf{y})^\top & \text{diag}(\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \end{bmatrix}, \\ \text{Hess}(\tilde{f} - f)(\mathbf{x}, \mathbf{y}) &= \frac{\varepsilon}{ne^B} \begin{bmatrix} \text{diag}(e^x + e^{-x}) & \mathbf{0} \\ \mathbf{0} & \text{diag}(e^y + e^{-y}) \end{bmatrix}. \end{aligned} \quad (15.2.3)$$

Note that computing $\text{Hess } \tilde{f}(\mathbf{x}, \mathbf{y})$ up to high precision can be done using $\tilde{O}(m)$ classical queries to \mathbf{A} , \mathbf{x} , and \mathbf{y} . Below we show how to obtain a sparse approximation of $\text{Hess } \tilde{f}(\mathbf{x}, \mathbf{y})$ using only $\tilde{O}(\sqrt{mn})$ quantum queries. We will do so in the sense of condition (2) of Theorem 15.1.5 where we take \mathbf{H}'_m to be a (high-precision) additive approximation of $\text{Hess } f(\mathbf{x}, \mathbf{y})$, and $\mathbf{H}'_a = \text{Hess } \tilde{f}(\mathbf{x}, \mathbf{y}) - \mathbf{H}'_m$.

We first obtain a multiplicative spectral approximation of (a high-precision additive approximation of) $\text{Hess } f(\mathbf{x}, \mathbf{y})$. In order to do so we use its structure: it is similar to a Laplacian matrix. This allows us to use the recent quantum Laplacian sparsifier of Apers and de Wolf (Theorem 13.3.5).

Lemma 15.2.2. *Given quantum query access to \mathbf{x}, \mathbf{y} and sparse quantum query access to \mathbf{A} , such that $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq C$, we can compute an SDD matrix \mathbf{H}_m with $\tilde{O}(n)$ non-zero entries, each off-diagonal entry non-negative, such that there exist symmetric \mathbf{H}'_m and $\mathbf{H}'_{a,f}$ satisfying $\mathbf{H}'_m + \mathbf{H}'_{a,f} = \text{Hess } f(\mathbf{x}, \mathbf{y})$, and*

$$0.9\mathbf{H}_m \leq \mathbf{H}'_m \leq 1.1\mathbf{H}_m, \quad \|\mathbf{H}'_{a,f}\|_1 \leq \delta_a,$$

in time $\tilde{O}(\sqrt{mn} \text{polylog}(C/\delta_a))$.

Proof. The key observation is that $\text{Hess } f(\mathbf{x}, \mathbf{y})$ satisfies

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} \end{bmatrix} \text{Hess } f(\mathbf{x}, \mathbf{y}) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} \end{bmatrix} = \begin{bmatrix} \text{diag}(\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) & -\mathbf{A}(\mathbf{x}, \mathbf{y}) \\ -\mathbf{A}(\mathbf{x}, \mathbf{y})^\top & \text{diag}(\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \end{bmatrix},$$

which is the Laplacian of the bipartite graph whose bipartite adjacency matrix is given by $\mathbf{A}(\mathbf{x}, \mathbf{y})$. Any off-diagonal entry of \mathbf{H} can be computed with additive error $\delta_a/2(2m+2n)$ using a single query to \mathbf{A} , to \mathbf{x} and to \mathbf{y} : the (i, j) -th entry of $\mathbf{A}(\mathbf{x}, \mathbf{y})$ is $A_{ij}e^{x_i+y_j}$, which is at most C (since $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq C$ by assumption), so we can compute $e^{x_i+y_j}$ to sufficient precision ($\lceil \log_2(C/\mu) \rceil + O(1)$ leading bits and $\lceil \log_2(2(2m+2n)/\delta_a) \rceil + O(1)$ trailing bits) and multiply it with A_{ij} . We can do this in such a way that if $A_{ij} = 0$, then the resulting entry is 0, and such that the approximation of $A_{ij}e^{x_i+y_j}$ is always non-negative.

Let \mathbf{H}' be the matrix whose off-diagonal entries are given by these approximations of the corresponding entries of \mathbf{H} , and whose diagonal entries are such that \mathbf{H}' is Laplacian. Then $\|\mathbf{H}' - \mathbf{H}\|_1 \leq \delta_a$ by the chosen precision for the additive approximation. Furthermore, as described before, a single query to off-diagonal entries of \mathbf{H}' can be implemented using a single query to \mathbf{A} , \mathbf{x} and \mathbf{y} . We may then use Theorem 13.3.5 to sparsify \mathbf{H}' , which uses $\tilde{O}(\sqrt{mn})$ queries to the off-diagonal entries of \mathbf{H}' and outputs a 0.1-spectral sparsification $\tilde{\mathbf{H}}$ of \mathbf{H}' that has $\tilde{O}(n)$ non-zero entries. Note that every non-zero entry of $\tilde{\mathbf{H}}$ was already non-zero in \mathbf{H}' because it is the Laplacian of a reweighted subgraph of the graph described by \mathbf{H}' ; hence any non-zero off-diagonal entry in $\tilde{\mathbf{H}}$ is contained in either the upper right or lower left $n \times n$ block, and each such entry is non-positive. Then the matrix $\mathbf{H}_m = \text{diag}(\mathbf{I}, -\mathbf{I})\tilde{\mathbf{H}}\text{diag}(\mathbf{I}, -\mathbf{I})$ satisfies the conclusion in the lemma, with $\mathbf{H}'_m = \text{diag}(\mathbf{I}, -\mathbf{I})\mathbf{H}'\text{diag}(\mathbf{I}, -\mathbf{I})$ and $\mathbf{H}'_{a,f} = \text{diag}(\mathbf{I}, -\mathbf{I})(\mathbf{H} - \mathbf{H}')\text{diag}(\mathbf{I}, -\mathbf{I})$. \square

We now show how to compute an additive approximation of the Hessian of the regularization term in \tilde{f} .

Lemma 15.2.3. *Given quantum query access to \mathbf{x}, \mathbf{y} with $\|\mathbf{x}\|_\infty, \|\mathbf{y}\|_\infty \leq B + \ln(4n + (n \ln(1/\mu)/\epsilon))$, we can compute a non-negative diagonal matrix $\mathbf{H}_{a,\tilde{f}}$ that satisfies $\|\mathbf{H}_{a,\tilde{f}} - \text{Hess}(\tilde{f} - f)(\mathbf{x}, \mathbf{y})\|_1 \leq \delta_a$, in time $\tilde{O}(n \log(1/\delta_a \mu) \text{polylog}(\epsilon))$.*

Proof. Recall that $\text{Hess}(\tilde{f} - f)(\mathbf{x}, \mathbf{y})$ is a diagonal matrix whose entries are of the form $\frac{\varepsilon}{ne^B}(e^{x_i} + e^{-x_i})$ or $\frac{\varepsilon}{ne^B}(e^{y_i} + e^{-y_i})$. Note that by assumption on the ℓ^∞ -norms of \mathbf{x} and \mathbf{y} , all diagonal entries are upper bounded by

$$2 \frac{\varepsilon}{ne^B} e^{B + \ln(4n + (n \ln(1/\mu)/\varepsilon))} = 8\varepsilon + 2 \ln(1/\mu).$$

Hence, it suffices to compute each diagonal entry using $\lceil \log_2(8\varepsilon + 2 \ln(1/\mu)) \rceil$ leading bits and $\lceil \log_2(1/n\delta_a) \rceil$ trailing bits. We can do so efficiently by using the identity

$$\frac{\varepsilon}{ne^B}(e^{x_i} + e^{-x_i}) = \frac{\varepsilon}{n}(e^{x_i - B} + e^{-x_i - B})$$

and the analogous one for y_i . \square

Theorem 15.2.4. *Given quantum query access to \mathbf{x}, \mathbf{y} with $\|\mathbf{x}\|_\infty, \|\mathbf{y}\|_\infty \leq B + \ln(4n + (n \ln(1/\mu)/\varepsilon))$, and sparse quantum query access to \mathbf{A} , if $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq C$, then we can compute (classical descriptions of) an SDD matrix \mathbf{H}_m with $\tilde{O}(n)$ non-zero entries, with all of the off-diagonal entries non-negative, and a non-negative diagonal matrix \mathbf{H}_a such that there exist symmetric $\mathbf{H}'_m, \mathbf{H}'_a$ with $\mathbf{H}'_m + \mathbf{H}'_a = \text{Hess} \tilde{f}(\mathbf{x}, \mathbf{y})$ and*

$$0.9\mathbf{H}_m \leq \mathbf{H}'_m \leq 1.1\mathbf{H}_m, \quad \|\mathbf{H}_a - \mathbf{H}'_a\|_1 \leq \delta_a$$

in quantum time $\tilde{O}(\sqrt{mn} \text{polylog}(C/\mu\delta_a))$.

Proof. Let \mathbf{H}_m be the matrix obtained from Lemma 15.2.2, and let \mathbf{H}_a be the matrix $\mathbf{H}_{a,\tilde{f}}$ obtained from Lemma 15.2.3, with precision $\delta_a/2$. Then \mathbf{H}_m and \mathbf{H}_a satisfy the desired properties, with \mathbf{H}'_m as in Lemma 15.2.2, and $\mathbf{H}'_a = \mathbf{H}'_{a,f} + \text{Hess}(\tilde{f} - f)(\mathbf{x}, \mathbf{y})$ with $\mathbf{H}'_{a,f}$ as in Lemma 15.2.2. \square

Now we compute a good approximation of the gradient of \tilde{f} , which is given by

$$\text{grad } \tilde{f}(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} r_1(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_1 \\ \vdots \\ r_n(\mathbf{A}(\mathbf{x}, \mathbf{y})) - r_n \\ c_1(\mathbf{A}(\mathbf{x}, \mathbf{y})) - c_1 \\ \vdots \\ c_n(\mathbf{A}(\mathbf{x}, \mathbf{y})) - c_n \end{bmatrix} + \frac{\varepsilon}{ne^B} \begin{bmatrix} e^{x_1} - e^{-x_1} \\ \vdots \\ e^{x_n} - e^{-x_n} \\ e^{y_1} - e^{-y_1} \\ \vdots \\ e^{y_n} - e^{-y_n} \end{bmatrix},$$

We can use quantum approximate summing to compute the row and column marginals with multiplicative error $1 \pm \delta$, which translates into additive error $\delta \cdot r_i(\mathbf{A}(\mathbf{x}, \mathbf{y}))$ or similar for the column sums. This yields an ℓ^1 -error in the vector of row or column sums which scales with $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$; see Corollary 13.3.1 for details. The part of the gradient coming from the regularization term is dealt with similarly as in Lemma 15.2.3.

Lemma 15.2.5. *Given quantum query access to (b_1, b_2) -fixed-point representations of $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and sparse quantum query access to rational $\mathbf{A} \in [0, 1]^{n \times n}$, if $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \leq C$, we can find a classical description of a vector $\mathbf{b} \in \mathbb{R}^n$ such that*

$$\|\mathbf{b} - \text{grad } \tilde{f}(\mathbf{x}, \mathbf{y})\|_1 \leq \delta \cdot C$$

in quantum time $\tilde{O}(\sqrt{mn}/\delta \cdot \text{polylog}(C/\mu))$, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1, b_2 and the encoding length of \mathbf{A} , and polylogarithmic factors in n and $1/\delta$.

15.2.3. Quantum box-constrained matrix scaling

Combining the quantum algorithms for Hessian and gradient estimation (Theorem 15.2.4 and Lemma 15.2.5) with the general framework of optimizing second-order robust functions (Theorem 15.1.5), we can obtain a quantum algorithm for matrix scaling that is based on classical box-constrained Newton methods. See Algorithm 15.1 for its formal definition.

Before analyzing the algorithm, we show that throughout our algorithm, $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ is bounded above by a constant; if $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$ is too large, we can change the overall scaling of the matrix and decrease the regularized potential (so in particular, we stay in the sublevel set of the regularized potential).

Lemma 15.2.6. *Let \mathbf{x}, \mathbf{y} be such that $\tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0}, \mathbf{0})$, and assume $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \geq C'$ where $C' > 1$. Let $\mathbf{x}' = \mathbf{x} - \ln(\gamma)\mathbf{1}$ where $1 \leq \gamma \leq C'$. Then*

$$\tilde{f}(\mathbf{x}', \mathbf{y}) - \tilde{f}(\mathbf{x}, \mathbf{y}) \leq \left(\frac{1}{\gamma} - 1\right)C' + \ln(\gamma) + (\gamma - 1) \left(\ln(1/\mu) + \frac{4\varepsilon}{e^B} \right)$$

Proof. We have

$$\begin{aligned} & \tilde{f}(\mathbf{x}', \mathbf{y}) - \tilde{f}(\mathbf{x}, \mathbf{y}) \\ &= \left(\frac{1}{\gamma} - 1\right)\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 + \langle \mathbf{r}, \ln(\gamma)\mathbf{1} \rangle + \frac{\varepsilon}{ne^B} \left(\sum_i (e^{x_i - \ln(\gamma)} - e^{x_i} + e^{-x_i + \ln(\gamma)} - e^{-x_i}) \right) \\ &= \left(\frac{1}{\gamma} - 1\right)\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 + \ln(\gamma) + \frac{\varepsilon}{ne^B} \left(\frac{1}{\gamma} - 1\right) \left(\sum_i e^{x_i} \right) + \frac{\varepsilon}{ne^B} (\gamma - 1) \left(\sum_i e^{-x_i} \right) \\ &\leq \left(\frac{1}{\gamma} - 1\right)\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 + \ln(\gamma) + 0 + \frac{\varepsilon}{ne^B} (\gamma - 1) \left(\sum_i e^{-x_i} \right) \\ &\leq \left(\frac{1}{\gamma} - 1\right)C' + \ln(\gamma) + (\gamma - 1) \left(\ln(1/\mu) + \frac{4\varepsilon}{e^B} \right) \end{aligned}$$

where for the last inequality we use $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \geq C'$ for the first term and Eq. (15.2.2) for the last term. \square

An appropriate choice of C' and γ makes the bound in the above lemma non-positive.

Corollary 15.2.7. *Let $\varepsilon \leq 1$ and $\mu \leq 1$, set $\gamma = 2$ and $C' = 2(\ln(2/\mu) + 4\varepsilon/e^B)$. Then, if $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1 \geq C'$ and $\tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0}, \mathbf{0})$, we have $\tilde{f}(\mathbf{x}', \mathbf{y}) \leq \tilde{f}(\mathbf{x}, \mathbf{y})$.*

Now we are ready to analyze Algorithm 15.1.

Theorem 15.2.8. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ with m non-zero entries, $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{>0}^n$ such that $\|\mathbf{r}\|_1 = 1 = \|\mathbf{c}\|_1$, and assume \mathbf{A} is asymptotically (\mathbf{r}, \mathbf{c}) -scalable. Let $\varepsilon > 0$, let $B \geq 1$, and assume there exist $(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)$ such that $\|(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)\|_\infty \leq B$ and $f(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon) - f^* \leq \varepsilon$. Furthermore, let \mathcal{A} be the $O(\log(n))$ -oracle of Theorem 15.1.4. Then Algorithm 15.1 with these parameters outputs, with probability $\geq 2/3$, vectors \mathbf{x}, \mathbf{y} such that $f(\mathbf{x}, \mathbf{y}) - f^* \leq 6\varepsilon$ and runs in quantum time $\tilde{O}(B^{1.5} \sqrt{mn}/\varepsilon)$.*

Algorithm 15.1: Quantum box-constrained Newton method for matrix scaling

Input: Query access to $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and smallest non-zero entry $\mu > 0$, error $\varepsilon > 0$, targets $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{>0}^n$ with $\|\mathbf{r}\|_1 = 1 = \|\mathbf{c}\|_1$, diameter bound $B \geq 1$, classical k -oracle \mathcal{A} for SDD matrices with non-negative off-diagonal entries

Output: Vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with $\|(\mathbf{x}, \mathbf{y})\|_\infty \leq B + \ln(4n + (n \ln(1/\mu)/\varepsilon))$

Analysis: Theorem 15.2.8 and Corollaries 15.2.9 and 15.2.10

```

1 set  $T = \lceil 4e^4 \max(k(B + \ln(4n + (n \ln(1/\mu)/\varepsilon))), 1) \cdot \ln\left(\frac{\ln(1/\mu) + 4\varepsilon/e^B}{\varepsilon/2}\right) \rceil$ ;
2 set  $C' = 2\lceil \ln(2/\mu) + 8\varepsilon/e^B \rceil$ ;
3 set  $\varepsilon' = \lfloor \varepsilon/8e^4 \max(k(B + \ln(4n + (n \ln(1/\mu)/\varepsilon))), 1) \rfloor$ ;
4 store  $\mathbf{x}^{(0)}, \mathbf{y}^{(0)} = \mathbf{0} \in \mathbb{R}^n$  in QCRAM;
5 for  $i = 0, \dots, T - 1$  do
6   compute  $\mathbf{H}_m, \mathbf{H}_a$  s.t.  $\mathbf{H}_m + \mathbf{H}_a \approx \text{Hess } \tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$  as in Theorem 15.2.4
   with  $\delta_a = \varepsilon'/2e^2$ ;
7   compute  $\mathbf{b} \approx \text{grad } \tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$  as in Lemma 15.2.5 at  $\mathbf{x}^{(i)}, \mathbf{y}^{(i)}$  with
    $\delta = \varepsilon'/3$ ;
8   compute  $\Delta = \mathcal{A}(\frac{4e^2}{3k^2} \cdot (\mathbf{H}_m + \mathbf{H}_a), \frac{\mathbf{b}}{k})$ ;
9   compute  $(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)}) = (\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) + \frac{1}{k}\Delta$  and store in QCRAM;
10  set flag = true;
11  while flag do
12    Compute  $C'/2$ -additive approximation  $\gamma$  of  $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1$ ;
13    if  $\gamma \leq 3C'/2$  then
14      | set flag = false;
15    end if
16    else
17      | update  $\mathbf{x}^{(i+1)} \leftarrow \mathbf{x}^{(i+1)} - \ln(2)\mathbf{1}$  in QCRAM;
18    end if
19  end while
20 end for
21 return  $(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^{(T)}, \mathbf{y}^{(T)})$ ;

```

Proof. In every iteration, the matrices $\mathbf{H}_m, \mathbf{H}_a$ and the vector \mathbf{b} are such that they satisfy the requirements of Theorem 15.1.5, hence

$$\tilde{f}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)}) - \tilde{f}^* \leq \left(1 - \frac{1}{4e^4 \max(kR_\infty, 1)}\right) (\tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \tilde{f}^*) + e^2\delta_a + \frac{3\delta}{2}$$

where $R_\infty \leq B + \ln(4n + (n \ln(1/\mu)/\varepsilon))$ is the ℓ^∞ -radius of the sublevel set $\{(\mathbf{x}, \mathbf{y}) : \tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0}, \mathbf{0})\}$ about $(\mathbf{0}, \mathbf{0})$, whose upper bound follows from Lemma 15.2.1. From here on, we write $M = 4e^4 \max(kR_\infty, 1)$. The choice of δ_a and δ in the algorithm is such that $e^2\delta_a + 3\delta/2 \leq \frac{\varepsilon}{2M}$, hence we can also bound the progress by

$$\tilde{f}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)}) - \tilde{f}^* \leq \left(1 - \frac{1}{M}\right) (\tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \tilde{f}^*) + \frac{\varepsilon}{2M}.$$

Corollary 15.2.7 shows that if $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1$ is larger than C' , then we can shift \mathbf{x} by $-\ln(2)\mathbf{1}$, this halves $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1$ and does not increase the regularized potential. Repeating this roughly $\log_2(\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1/C')$ many times¹ reduces $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1$ to at most $C = 2C'$. Determining when to stop this process requires a procedure to distinguish between the cases $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1 \leq C'$ and $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1 \geq 2C'$ (if in between C' and $2C'$ either continuing or stopping is fine). Such a procedure can be implemented by computing a $C'/2$ -additive approximation of $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1$, which can be done using $\tilde{O}(\sqrt{mn} \text{polylog}(C'/\mu))$ quantum queries (Lemma 13.3.9). Therefore, throughout the algorithm we may assume that $\|\mathbf{A}(\mathbf{x}^{(i+1)}, \mathbf{y}^{(i+1)})\|_1 \leq 2C' = C$.

It remains to show that $\tilde{f}(\mathbf{x}^{(T)}, \mathbf{y}^{(T)}) - \tilde{f}^* \leq \varepsilon$ for our choice of T . Note that we have

$$\begin{aligned} \tilde{f}(\mathbf{x}^{(T)}, \mathbf{y}^{(T)}) - \tilde{f}^* &\leq \left(1 - \frac{1}{M}\right)^T (\tilde{f}(\mathbf{0}, \mathbf{0}) - \tilde{f}^*) + \sum_{i=0}^{T-1} \left(1 - \frac{1}{M}\right)^{T-i-1} \cdot \frac{\varepsilon}{2M} \\ &\leq \left(1 - \frac{1}{M}\right)^T (\tilde{f}(\mathbf{0}, \mathbf{0}) - \tilde{f}^*) + \left(1 - \left(1 - \frac{1}{M}\right)^T\right) \cdot \frac{\varepsilon}{2} \\ &\leq \left(1 - \frac{1}{M}\right)^T (f(\mathbf{0}, \mathbf{0}) - f^* + \frac{2\varepsilon}{e^B}) + \frac{\varepsilon}{2} \\ &\leq \left(1 - \frac{1}{M}\right)^T (\ln(1/\mu) + \frac{2\varepsilon}{e^B}) + \frac{\varepsilon}{2} \\ &\leq \varepsilon \end{aligned}$$

where the third inequality uses $\tilde{f} \geq f^*$ and $\tilde{f}(\mathbf{0}, \mathbf{0}) = f(\mathbf{0}, \mathbf{0}) + 4\varepsilon/e^B$ (Lemma 15.2.1), and in the last inequality we use

$$\begin{aligned} T &= \left\lceil 4e^4 \max(k(B + \ln(4n + (n \ln(1/\mu)/\varepsilon))), 1) \cdot \ln\left(\frac{\ln(1/\mu) + 4\varepsilon/e^B}{\varepsilon/2}\right) \right\rceil \\ &\geq \left\lceil M \cdot \ln\left(\frac{\ln(1/\mu) + 4\varepsilon/e^B}{\varepsilon/2}\right) \right\rceil \\ &\geq \frac{1}{\ln(1 - \frac{1}{M})} \cdot \ln\left(\frac{\varepsilon/2}{\ln(1/\mu) + \frac{4\varepsilon}{e^B}}\right). \end{aligned}$$

This implies that

$$f(\mathbf{x}^{(T)}, \mathbf{y}^{(T)}) - f^* \leq \tilde{f}(\mathbf{x}^{(T)}, \mathbf{y}^{(T)}) - \tilde{f}^* + 5\varepsilon \leq 6\varepsilon,$$

where we crucially use the last point of Lemma 15.2.1 and the assumption that there exist $(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)$ with $\|(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)\|_\infty \leq B$ which ε -minimize f .

Finally we bound the time complexity of Algorithm 15.1. For each of the quoted results, we use the choice $C = 2C' = \tilde{O}(\ln(n) + \varepsilon)$. In each of the T iterations we compute:

- (i) approximations $\mathbf{H}_m, \mathbf{H}_a$ of Hess $\tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ in time $\tilde{O}(\sqrt{mn} \text{polylog}(1/\varepsilon))$ (Theorem 15.2.4, using that $C, 1/\mu$ are at most $\text{poly}(n)$),

¹Which is an almost constant number of times: in a single update of the box-constrained method, we take steps of size at most 1 in ℓ^∞ -norm, so individual entries can only grow by a factor e^2 in a single iteration, and the holds same for $\|\mathbf{A}(\mathbf{x}, \mathbf{y})\|_1$.

- (ii) an $\varepsilon'/3$ - ℓ^1 -approximation of $\nabla \tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ in time $\tilde{O}(\sqrt{mn/\varepsilon'}) = \tilde{O}(\sqrt{B mn/\varepsilon})$ (Lemma 15.2.5)
- (iii) an update Δ in time $\tilde{O}(n)$ using one call to the $k = O(\log(n))$ -oracle on SDD-matrices with $\tilde{O}(n)$ non-zero entries (Theorem 15.1.4)
- (iv) at most $O(1)$ many times (using the fact that in Line 9 the 1-norm changes by at most a constant factor since $\|\frac{1}{k}\Delta\|_\infty \leq 1$) an $O(\ln(1/\mu) + \varepsilon)$ -additive approximation of $\|\mathbf{A}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\|_1$ in time $\tilde{O}(\sqrt{mn})$ (Lemma 13.3.9).

Note that the second contribution dominates the others, resulting in an overall time complexity $\tilde{O}(B^{1.5}\sqrt{mn/\varepsilon})$. \square

The above theorem assumes that a bound B on the ℓ^∞ -norm of an ε -minimizer of f is known. For the purpose of matrix scaling, one can circumvent this assumption by running the algorithm for successive powers of 2 (i.e., $B = 1, B = 2, B = 4, \dots$) and testing after each run whether the output provides an ε -relative-entropy-scaling or not. Verifying whether given \mathbf{x}, \mathbf{y} provide an ε -relative-entropy-scaling of \mathbf{A} can be done in time $\tilde{O}(\sqrt{mn/\varepsilon})$ (see Theorem 13.3.3). Note that this gives an algorithm for ε -relative-entropy-scaling whose complexity depends on a diameter bound for ε -minimizers of f , rather than a diameter bound for ε -relative-entropy-scaling vectors. Furthermore, such an iterative doubling approach does not work for the task of finding an ε -minimizer of f , as we do not know how to test this property efficiently.

As a consequence of Theorem 15.2.8 and Lemma 14.1.2, we deduce the following result.

Corollary 15.2.9. *For asymptotically-scalable matrices $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ with m non-zero entries, there is a quantum algorithm that finds ε -relative-entropy-scaling vectors (\mathbf{x}, \mathbf{y}) of \mathbf{A} to target marginals $\mathbf{r}, \mathbf{c} \in \mathbb{R}_{> 0}^n$ with $\|\mathbf{r}\|_1 = 1 = \|\mathbf{c}\|_1$ in time $\tilde{O}(R_\infty^{1.5}\sqrt{mn/\varepsilon})$, where R_∞ is such that there exists an ε -approximate minimizer $(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)$ of f with*

$$R_\infty = \|(\mathbf{x}_\varepsilon, \mathbf{y}_\varepsilon)\|_\infty + \ln(4n + (n \ln(1/\mu)/\varepsilon)).$$

We can use Pinsker's inequality (Lemma 13.2.1) to convert the above to a result about ℓ^1 -scaling, yielding a ε - ℓ^1 -scaling in time $\tilde{O}(R_\infty^{1.5}\sqrt{mn/\varepsilon})$. Note that the relevant radius bound R_∞ is that of an $O(\varepsilon^2)$ -approximate minimizer of f instead of an $O(\varepsilon)$ -approximate minimizer.

For the general case mentioned above, we do not have good (i.e., polylogarithmic) bounds on the parameter R_∞ . We do have such bounds when \mathbf{A} is entrywise positive: it is well-known (and easy to show²) that such an \mathbf{A} can be exactly scaled to

²From the inequality $A_{ij}e^{x_i+y_j} \leq 1/n$ one gets the upper bounds $x_i + y_j \leq \ln(1/n\mu)$ for every i, j . To obtain a variation norm bound for \mathbf{x} and \mathbf{y} , note that for every fixed i , there is at least one j_i such that $A_{ij_i}e^{x_i+y_{j_i}} \geq 1/n^2$ (because the row sums are $1/n$). Therefore $x_i + y_{j_i} \geq \ln(1/n^2\nu)$ where ν is the largest entry of \mathbf{A} , and $x_{i'} - x_i = (x_{i'} + y_{j_i}) - (x_i + y_{j_i}) \leq \ln(1/n\mu) - \ln(1/n^2\nu) = \ln(n\nu/\mu)$ for every i, i' . This is an upper bound on the variation norm of \mathbf{x} , and one can derive the same bound for that of \mathbf{y} . By translating \mathbf{x}, \mathbf{y} by appropriate multiples of the all-ones vector we can assume $x_1 = 0$. Then the variation-norm bound also bounds the ℓ^∞ -norm of \mathbf{x} . To then get an ℓ^∞ -bound on \mathbf{y} , recall that for at least one j , one has $x_1 + y_j = y_j \geq \ln(1/n^2\nu) \geq 0$ and still $y_j = x_1 + y_j \leq \ln(1/n\mu)$, so $\|\mathbf{y}\|_\infty \leq \ln(1/n\mu) + \ln(n\nu/\mu) = \ln(\nu/\mu^2)$.

uniform marginals with scaling vectors (\mathbf{x}, \mathbf{y}) such that $\|(\mathbf{x}, \mathbf{y})\|_\infty = O(\log(\|\mathbf{A}\|_1/\mu))$ ([KK96, Lem. 1], [CMTV17, Lem. 4.11]). In particular, this implies that there exists a minimizer $(\mathbf{x}^*, \mathbf{y}^*)$ of f with $\|(\mathbf{x}^*, \mathbf{y}^*)\|_\infty = O(\log(\|\mathbf{A}\|_1/\mu)) = \tilde{O}(1)$ and therefore we have the following corollary.

Corollary 15.2.10. *For entrywise-positive matrices \mathbf{A} , there is a quantum algorithm that finds an ε -relative-entropy-scaling of \mathbf{A} to uniform marginals in time $\tilde{O}(n^{1.5}/\sqrt{\varepsilon})$. Similarly, it finds an ε - ℓ^1 -scaling of \mathbf{A} to uniform marginals in time $\tilde{O}(n^{1.5}/\varepsilon)$.*

15.2.4. Optimality of the choice of parameters.

A natural question is whether the current choice of approximation precision in every iteration is essentially optimal: when the potential gap is large, the loss in potential decrease due to imprecision in the gradient estimation is less relevant. Therefore one may hope that a more dynamic choice of precision yields a better complexity. We show that this is not the case: using the same precision in every iteration is essentially optimal.

To formalize the argument, we proceed as follows. Let $z_i = \tilde{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \tilde{f}^*$ be the regularized potential gap in the i -th iteration. Then the z_i satisfy constraints of the following form:

$$z_{i+1} \leq (1 - \gamma)z_i + \delta_i,$$

where $\gamma = \frac{1}{4e^4 \max(kR_\infty, 1)}$ and δ_i is a parameter that determines the accuracy with which we approximate the gradient and Hessian in each iteration. In the algorithm we used the choice $\delta_i = e^2 \delta_a + \frac{3\delta}{2}$, independently of i . Since the complexity in each iteration scales as $1/\sqrt{\delta_i}$ as a function of δ_i , a natural question is whether one can obtain a better overall complexity by letting δ_i depend on i . In the following lemma we show this is not the case.

Lemma 15.2.11. *Let $z_0 > 0$, $\varepsilon > 0$ and $0 < \gamma \leq 1/2$ be given. Then, for any $N \geq 1$ and any choice of sequence of $\delta_0, \dots, \delta_{N-1} > 0$ such that the sequence defined by*

$$z_{i+1} = (1 - \gamma)z_i + \delta_i, \quad 0 \leq i \leq N - 1$$

satisfies $z_N \leq \varepsilon$, one must have

$$\sum_{i=0}^{N-1} \frac{1}{\sqrt{\delta_i}} \geq \frac{1}{\gamma^{3/2} \sqrt{\varepsilon}} \left(1 - (\varepsilon/z_0)^{1/3}\right)^{3/2}.$$

Proof. Observe that we have the explicit expression

$$z_N = (1 - \gamma)^N z_0 + \sum_{i=0}^{N-1} (1 - \gamma)^{N-i-1} \delta_i.$$

As every term in this sum is positive, we must have $(1 - \gamma)^N z_0 < \varepsilon$ if $z_N \leq \varepsilon$ (where we have strict inequality since $N \geq 1$ and therefore the sum is not empty). Now fix N such that $(1 - \gamma)^N z_0 < \varepsilon$, and define the Lagrangian $L(\delta_0, \dots, \delta_{N-1}; \lambda)$ by

$$L(\delta_0, \dots, \delta_{N-1}; \lambda) = \sum_{i=0}^{N-1} \frac{1}{\sqrt{\delta_i}} + \lambda \left((1 - \gamma)^N z_0 + \sum_{i=0}^{N-1} (1 - \gamma)^{N-i-1} \delta_i - \varepsilon \right).$$

Observe that the Lagrangian is convex in the δ_i and that the constraint $z_N \leq \varepsilon$ is linear in the δ_i , and can be made strict for a very small choice of δ_i . This shows that the Karush–Kuhn–Tucker conditions are satisfied, so that $\sum_{i=0}^{N-1} 1/\sqrt{\delta_i}$ is minimized subject to the constraint $z_N \leq \varepsilon$ if and only if $\text{grad } L(\delta_0, \dots, \delta_{N-1}; \lambda) = 0$ for some $\lambda \geq 0$. This gradient vanishes if and only if

$$-\frac{1}{2} \frac{1}{\delta_i^{3/2}} + \lambda \cdot (1-\gamma)^{N-i-1} = 0, \quad 0 \leq i \leq N-1, \quad \varepsilon = (1-\gamma)^N z_0 + \sum_{i=0}^{N-1} (1-\gamma)^{N-i-1} \delta_i.$$

For fixed $\lambda > 0$ this means that the optimal choice of δ_i is

$$\delta_i = (2\lambda)^{-2/3} \left((1-\gamma)^{-2/3} \right)^{N-i-1} = c_\lambda ((1-\gamma)^{-2/3})^{N-i-1}$$

where $c_\lambda := (2\lambda)^{-2/3}$. The constraint $z_N = \varepsilon$ then allows us to express c_λ in terms of ε , γ , and z_0 :

$$\varepsilon - (1-\gamma)^N z_0 = c_\lambda \sum_{i=0}^{N-1} ((1-\gamma)^{1/3})^{N-i-1} = c_\lambda \cdot \frac{1 - (1-\gamma)^{N/3}}{1 - (1-\gamma)^{1/3}}.$$

This leads to an associated cost of

$$\sum_{i=0}^{N-1} \frac{1}{\sqrt{\delta_i}} = \frac{1}{\sqrt{c_\lambda}} \cdot \frac{1 - (1-\gamma)^{N/3}}{1 - (1-\gamma)^{1/3}} = \left(\frac{1 - (1-\gamma)^{N/3}}{1 - (1-\gamma)^{1/3}} \right)^{3/2} \cdot \frac{1}{\sqrt{\varepsilon - (1-\gamma)^N z_0}}.$$

As $\gamma \leq 1$ we have $1 - (1-\gamma)^{1/3} \leq \gamma$, and because $(1-\gamma)^N z_0 < \varepsilon$, we have

$$1 - (1-\gamma)^{N/3} > 1 - (\varepsilon/z_0)^{1/3}$$

and the cost satisfies

$$\sum_{i=0}^{N-1} \frac{1}{\sqrt{\delta_i}} \geq \left(\frac{1 - (\varepsilon/z_0)^{1/3}}{\gamma} \right)^{3/2} \cdot \frac{1}{\sqrt{\varepsilon - (1-\gamma)^N z_0}} \geq \frac{1}{\gamma^{3/2} \sqrt{\varepsilon}} \left(1 - (\varepsilon/z_0)^{1/3} \right)^{3/2}. \quad \square$$

15.3. Quantum box-constrained matrix balancing

We can use the same techniques to tackle the matrix balancing problem. Again, we present in the following order: (1) a regularized potential function for matrix balancing, (2) quantum algorithms for approximating the Hessian and gradient, and (3) a quantum box-constrained matrix balancing algorithm and its analysis. There is a non-trivial technical difference between this analysis and that for scaling: we are looking for *multiplicative* minimizers of the potential function f (as defined in Section 14.4), rather than additive minimizers, in light of Corollary 14.4.3. Instead, one could also consider running the box-constrained Newton method for (a regularized version of) the logarithmic potential $\log f$, but its Hessian is no longer symmetric diagonally-dominant (although it is still second-order robust). Naively, this would also affect the required precision for the gradient of f in each iteration; to circumvent this, we appeal to the natural multiplicative nature of our quantum summing and hence gradient estimation routine (see Lemma 15.3.6).

15.3.1. A second-order robust potential function for matrix balancing

Given a matrix $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ with smallest non-zero entry μ and zero diagonal, a desired error $\varepsilon > 0$, and some number $B > 0$, we consider the regularized potential function $\tilde{f}(\mathbf{x})$ given by

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}) + \frac{\mu\varepsilon}{ne^B} \left(\sum_i (e^{x_i} + e^{-x_i}) \right), \quad (15.3.1)$$

where $f(\mathbf{x}) = \|\mathbf{A}(\mathbf{x})\|_1 = \sum_{i,j=1}^n A_{ij} e^{x_i - x_j}$ is the commonly-used potential function for analyzing algorithms for matrix balancing (see Section 14.4), and we have regularized the potential similarly as in [CMTV17]. Note that compared to Section 15.2, there is an additional factor μ in the regularization term, and μ is a lower bound on $f^* = \inf_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$ when \mathbf{A} is asymptotically balanceable (Lemma 14.4.1). The following is an adaptation of [CMTV17, Lem. 4.23].

Lemma 15.3.1. *Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ be asymptotically balanceable. Let $\varepsilon > 0$ and $B > 0$ be given. Then*

- (i) \tilde{f} is second-order robust with respect to ℓ^∞ and its Hessian is SDD,
- (ii) $f(\mathbf{x}) \leq \tilde{f}(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^n$,
- (iii) for all $\mathbf{x} \in \mathbb{R}^n$ such that $\tilde{f}(\mathbf{x}) \leq \tilde{f}(\mathbf{0})$, we have $\|\mathbf{x}\|_\infty \leq B + \ln\left(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n\right)$, and
- (iv) for any \mathbf{x}_ε such that $f(\mathbf{x}_\varepsilon) \leq (1 + \varepsilon)f^*$ and $\|\mathbf{x}_\varepsilon\|_\infty \leq B$, we have $\tilde{f}(\mathbf{x}_\varepsilon) \leq (1 + 3\varepsilon)f^*$. In particular, if such a \mathbf{x}_ε exists, we have $|f^* - \tilde{f}^*| \leq 3\varepsilon f^*$.

Proof. The first two items are again easy to verify. For the third, let $\mathbf{x} \in \mathbb{R}^n$ be such that $\tilde{f}(\mathbf{x}) \leq \tilde{f}(\mathbf{0})$. Then

$$\frac{\mu\varepsilon}{ne^B} \left(\sum_{i=1}^n e^{x_i} + e^{-x_i} \right) \leq f(\mathbf{0}) - f(\mathbf{x}) + \frac{2\mu\varepsilon}{e^B} \leq \|\mathbf{A}\|_1 - \mu + \frac{2\mu\varepsilon}{e^B},$$

where the last inequality uses the fact that $f(\mathbf{0}) = \|\mathbf{A}\|_1$ and $f(\mathbf{x}) \geq f^* \geq \mu$ (Lemma 14.4.1). Thus we have $e^{x_i} + e^{-x_i} \leq \frac{ne^B(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n$ for every $i \in [n]$ and it follows that

$$|x_i| \leq \ln\left(\frac{ne^B(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n\right) \leq B + \ln\left(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n\right).$$

For the last item, note that if $\|\mathbf{x}_\varepsilon\|_\infty \leq B$, we have $e^{x_i} + e^{-x_i} \leq 2e^B$. Thus

$$\tilde{f}(\mathbf{x}_\varepsilon) \leq f(\mathbf{x}_\varepsilon) + 2\mu\varepsilon \leq f^* + 3\varepsilon f^*. \quad \square$$

15.3.2. Quantum algorithms for approximating the Hessian and gradient for balancing

In order to use Theorem 15.1.5 to minimize $\tilde{f}(\mathbf{x})$, we need to show how to approximate both the gradient and Hessian of $\tilde{f}(\mathbf{x})$. Before that, we state a simple lemma on computing a multiplicative approximation to $\|\mathbf{A}(\mathbf{x})\|_1$, used to determine the arithmetic precision for the next steps:

Lemma 15.3.2. *Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix with m possibly non-zero entries, and assume \mathbf{A} is asymptotically balanceable with smallest non-zero entry $\mu > 0$. Given quantum query access to (b_1, b_2) -fixed-point representations of $\mathbf{x} \in \mathbb{R}^n$, sparse quantum query access to $\mathbf{A} \in [0, 1]^{n \times n}$, and rational $\eta > 0$, we can find with probability at least $1 - \eta$ a real number $C \geq 0$ such that $C \leq \min(1, \|\mathbf{A}(\mathbf{x})\|_1) \leq 2C$ in quantum time $\tilde{O}(\sqrt{m})$, where the $\tilde{O}(\cdot)$ hides polynomial factors in b_1, b_2 and the encoding length of \mathbf{A} , and polylogarithmic factors in n .*

Proof. Use $\text{LogSumExp}(\mathbf{A}, 1, \mathbf{x}\mathbf{1}^\top - \mathbf{1}\mathbf{x}^\top, \frac{1}{4}, b'_1, b'_2, \eta, \mu)$ to compute an additive $\frac{1}{4}$ -approximation of $\ln(\|\mathbf{A}(\mathbf{x})\|_1)$, where the $b'_i \geq b_i$ are large enough to represent values in $[\ln(\mu), 1]$. Here we think of \mathbf{A} and $\mathbf{x}\mathbf{1}^\top - \mathbf{1}\mathbf{x}^\top$ as vectors of length n^2 where \mathbf{A} has m possibly non-zero entries. We can then exponentiate this value, returning 1 if $\ln(\|\mathbf{A}(\mathbf{x})\|_1) \geq 0$. An additive $\frac{1}{4}$ -approximation of the logarithm is also precise enough to obtain $C \geq 0$ with $C \leq \|\mathbf{A}(\mathbf{x})\|_1 \leq 2C$. Note that $\|\mathbf{A}(\mathbf{x})\|_1 \geq \mu$ since \mathbf{A} is asymptotically balanceable, so there the result can be represented with small bit complexity. \square

It is convenient to use LogSumExp here since we are dealing with $\mathbf{A}(\mathbf{x})$ rather than \mathbf{A} itself, and the exponentials in the entries are carefully dealt with in LogSumExp , but in principle one could also rely on ApproxSum .

Next, we consider the Hessian of $\tilde{f}(\mathbf{x})$, which can be written as the sum of the Hessian of f and the Hessian of the regularizer $\tilde{f} - f$:

$$\begin{aligned} \text{Hess } f(\mathbf{x}) &= \text{diag}(\mathbf{r}(\mathbf{A}(\mathbf{x})) + \mathbf{c}(\mathbf{A}(\mathbf{x}))) - (\mathbf{A}(\mathbf{x}) + \mathbf{A}(\mathbf{x})^\top), \\ \text{Hess}(\tilde{f} - f)(\mathbf{x}) &= \frac{\mu\epsilon}{ne^B} \text{diag}(e^{\mathbf{x}} + e^{-\mathbf{x}}). \end{aligned} \quad (15.3.2)$$

Computing $\text{Hess } \tilde{f}(\mathbf{x})$ up to high precision can be done using $\tilde{O}(m)$ classical queries to \mathbf{A} and \mathbf{x} . Below we show how to obtain a sparse approximation of $\text{Hess } \tilde{f}(\mathbf{x})$ using only $\tilde{O}(\sqrt{mn})$ quantum queries.

Lemma 15.3.3. *Given quantum query access to \mathbf{x} and sparse quantum query access to \mathbf{A} , such that $\|\mathbf{A}(\mathbf{x})\|_1 \leq C$, we can compute an SDD matrix \mathbf{H}_m with $\tilde{O}(n)$ non-zero entries, each off-diagonal entry non-negative, such that there exist symmetric \mathbf{H}'_m and $\mathbf{H}'_{\alpha, f}$ satisfying $\mathbf{H}'_m + \mathbf{H}'_{\alpha, f} = \text{Hess } f(\mathbf{x})$, and*

$$0.9\mathbf{H}_m \leq \mathbf{H}'_m \leq 1.1\mathbf{H}_m, \quad \|\mathbf{H}'_{\alpha, f}\|_1 \leq \delta_\alpha,$$

in time $\tilde{O}(\sqrt{mn} \text{polylog}(C/\delta_\alpha))$.

We omit the proof, as it is exactly as in the matrix scaling case (Lemma 15.2.2), with the only change that the Hessian $\text{Hess } f(\mathbf{x})$ is already Laplacian (the off-diagonal entries have the correct sign).

Now we show how to compute an additive approximation of $\text{grad}^2 \tilde{f}$.

15. Quantum box-constrained Newton methods

Lemma 15.3.4. *Given quantum query access to \mathbf{x} with $\|\mathbf{x}\|_\infty \leq B + \ln(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n)$, we can compute a non-negative diagonal matrix $\mathbf{H}_{\alpha, \tilde{f}}$ that satisfies $\|\mathbf{H}_{\alpha, \tilde{f}} - \text{grad}^2(\tilde{f} - f)(\mathbf{x})\| \leq \delta_\alpha$ in time $\tilde{O}(n \log(1/\delta_\alpha) \text{polylog}(\varepsilon))$.*

Proof. Note that $\text{grad}^2(\tilde{f} - f)(\mathbf{x})$ is diagonal with entries upper bounded by

$$\frac{2\mu\varepsilon}{ne^B} e^{B + \ln(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n)} = 2(\|\mathbf{A}\|_1 - \mu) + 4\mu\varepsilon.$$

We can compute each entry separately by querying \mathbf{x} . More precisely, we obtain $\mathbf{H}_{\alpha, \tilde{f}}$ by computing each entry using $\lceil \log_2(2(\|\mathbf{A}\|_1 - \mu) + 4\mu\varepsilon) \rceil$ leading bits and $\lceil \log_2(1/n\delta_\alpha) \rceil$ trailing bits. The chosen parameters make sure that $\|\mathbf{H}_{\alpha, \tilde{f}} - \text{grad}^2(\tilde{f} - f)(\mathbf{x})\| \leq \delta_\alpha$. \square

Combining these two lemmas gives a quantum algorithm for approximating $\text{grad}^2 \tilde{f}$.

Theorem 15.3.5. *Given quantum query access to \mathbf{x} with $\|\mathbf{x}\|_\infty \leq B + \ln(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n)$, and sparse quantum query access to \mathbf{A} , if $\|\mathbf{A}(\mathbf{x})\|_1 \leq C$, then we can compute (classical descriptions of) an SDD matrix \mathbf{H}_m with $\tilde{O}(n)$ non-zero entries, with all of the off-diagonal entries non-negative, and a non-negative diagonal matrix \mathbf{H}_α such that there exist symmetric $\mathbf{H}'_m, \mathbf{H}'_\alpha$ with $\mathbf{H}'_m + \mathbf{H}'_\alpha = \text{Hess} \tilde{f}(\mathbf{x})$ and*

$$0.9\mathbf{H}_m \leq \mathbf{H}'_m \leq 1.1\mathbf{H}_m, \quad \|\mathbf{H}_\alpha - \mathbf{H}'_\alpha\|_1 \leq \delta_\alpha$$

in quantum time $\tilde{O}(\sqrt{mn} \text{polylog}(C/\delta_\alpha))$.

Proof. Let \mathbf{H}_m be the matrix obtained from Lemma 15.3.3, and let \mathbf{H}_α be the matrix $\mathbf{H}_{\alpha, \tilde{f}}$ obtained from Lemma 15.3.4, both computed with additive error $\delta_\alpha/2$. Then \mathbf{H}_m and \mathbf{H}_α satisfy the desired properties, with \mathbf{H}'_m as in Lemma 15.3.3, and $\mathbf{H}'_\alpha = \mathbf{H}'_{\alpha, f} + \text{Hess}(\tilde{f} - f)(\mathbf{x}, \mathbf{y})$ with $\mathbf{H}'_{\alpha, f}$ as in Lemma 15.3.3. \square

Next, we show how to efficiently compute $\text{grad} \tilde{f}$, which is given by

$$\text{grad} \tilde{f}(\mathbf{x}) = \mathbf{r}(\mathbf{A}(\mathbf{x})) - \mathbf{c}(\mathbf{A}(\mathbf{x})) + \frac{\mu\varepsilon}{ne^B} \begin{bmatrix} e^{x_1} - e^{-x_1} \\ \vdots \\ e^{x_n} - e^{-x_n} \end{bmatrix}.$$

We approximate the row sums and column sums up to a multiplicative error $(1 \pm \delta)$, using quantum approximate summing (more precisely, Corollary 13.3.1). The gradient of the regularization term can be dealt with similarly as in Lemma 15.3.4. This results in the following:

Lemma 15.3.6. *Let $C > 0$ be given. Given quantum query access to $(\mathbf{b}_1, \mathbf{b}_2)$ -fixed-point representations of $\mathbf{x} \in \mathbb{R}^n$ and sparse quantum query access to rational $\mathbf{A} \in [0, 1]^{n \times n}$, if $\|\mathbf{A}(\mathbf{x})\|_1 \leq C$, we can find a classical description of a vector $\mathbf{b} \in \mathbb{R}^n$ such that*

$$\|\mathbf{b} - \text{grad} \tilde{f}(\mathbf{x})\|_1 \leq \delta \|\mathbf{A}(\mathbf{x})\|_1$$

in quantum time $\tilde{O}(\sqrt{mn}/\delta \text{polylog}(C))$, where the $\tilde{O}(\cdot)$ hides polynomial factors in $\mathbf{b}_1, \mathbf{b}_2$ and the encoding length of \mathbf{A} , and polylogarithmic factors in n and $1/\delta$.

We note here that the upper bound being of the form $\delta \|\mathbf{A}(\mathbf{x})\|_1$ (as opposed to δC) is important in the analysis of the box-constrained Newton method, see Theorem 15.3.7.

15.3.3. Quantum box-constrained matrix balancing

We now combine the quantum algorithms for Hessian and gradient estimation (Theorem 15.3.5 and Lemma 15.3.6) with the general framework for optimizing second-order robust functions (Theorem 15.1.5), obtaining a quantum algorithm for matrix balancing that is based on classical box-constrained Newton methods. See Algorithm 15.2 for its formal definition. Its analysis is as follows.

Algorithm 15.2: Quantum box-constrained Newton method for matrix balancing

Input: Query access to $\mathbf{A} \in [0, 1]^{n \times n}$ with $\|\mathbf{A}\|_1 \leq 1$ and smallest non-zero entry $\mu > 0$, error $\varepsilon > 0$, diameter bound $B \geq 1$, classical k -oracle \mathcal{A} for SDD matrices with non-negative off-diagonal entries.

Output: A vector $\mathbf{x} \in \mathbb{R}^n$ with $\|\mathbf{x}\|_\infty \leq B + \ln\left(\frac{n(1-\mu)}{\mu\varepsilon} + 2n\right)$.

Analysis: Theorem 15.3.7 and Corollaries 15.3.8 and 15.3.9

```

1 set  $T = \left\lceil 8e^4 \max\left(k\left(B + \ln\left(\frac{n(1-\mu)}{\mu\varepsilon} + 2n\right)\right), 1\right) \ln\left(\frac{1-\mu+2\mu\varepsilon/e^B}{\mu\varepsilon}\right) \right\rceil$ ;
2 set  $C = 1 + 2\varepsilon/e^B$ ;
3 set  $\varepsilon' = \lfloor \varepsilon/8e^4 \max(k(B + \ln(\frac{n(1-\mu)}{\mu\varepsilon} + 2n)), 1) \rfloor$ ;
4 store  $\mathbf{x}^{(0)} = \mathbf{0} \in \mathbb{R}^n$  in QCRAM;
5 for  $i = 0, \dots, T - 1$  do
6   compute  $C' \geq 0$  such that  $C' \leq \|\mathbf{A}(\mathbf{x})\|_1 \leq 2C'$  using Lemma 15.3.2;
7   compute  $\mathbf{H}_m, \mathbf{H}_a$  s.t.  $\mathbf{H}_m + \mathbf{H}_a \approx \text{Hess } \tilde{f}(\mathbf{x}^{(i)})$  as in Theorem 15.3.5 with
    $\delta_a = \varepsilon' C' / 2e^2$ ;
8   compute  $\mathbf{b}$  with  $\|\mathbf{b} - \text{grad } \tilde{f}(\mathbf{x}^{(i)})\| \leq \delta \|\mathbf{A}(\mathbf{x}^{(i)})\|_1$  as in Lemma 15.3.6 at
    $\mathbf{x}^{(i)}$  with  $\delta = \varepsilon' / 3$ ;
9   compute  $\Delta = \mathcal{A}\left(\frac{4e^2}{3k^2} \cdot (\mathbf{H}_m + \mathbf{H}_a), \frac{\mathbf{b}}{k}\right)$ ;
10  compute  $\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \frac{1}{k}\Delta$  and store in QCRAM;
11 end for
12 return  $\mathbf{x} = \mathbf{x}^{(T)}$ ;
```

Theorem 15.3.7. Let $\mathbf{A} \in [0, 1]^{n \times n}$ be a rational matrix whose non-zero entries are at least $\mu > 0$, with only zeros on the diagonal and each row and column containing at least one non-zero element. We assume \mathbf{A} is asymptotically balanceable and $\|\mathbf{A}\|_1 \leq 1$. Let $\varepsilon \in (0, 1]$, $B \geq 1$, and assume there exists \mathbf{x}_ε such that $\|\mathbf{x}_\varepsilon\|_\infty \leq B$ and $f(\mathbf{x}_\varepsilon) \leq (1 + \varepsilon)f^*$. Furthermore, let \mathcal{A} be the $O(\log(n))$ -oracle of Theorem 15.1.4. Then Algorithm 15.2 with these parameters outputs, with probability $\geq 2/3$, a classical description of a vector $\mathbf{x} \in \mathbb{R}^n$ such that $f(\mathbf{x}) \leq (1 + 5\varepsilon + 6\varepsilon^2)f^*$ and runs in quantum time $\tilde{O}(B^{1.5}\sqrt{mn/\varepsilon})$.

Proof. In every iteration, the matrices $\mathbf{H}_m, \mathbf{H}_a$ and the vector \mathbf{b} are such that they satisfy the requirements of Theorem 15.1.5, hence

$$\tilde{f}(\mathbf{x}^{(i+1)}) - \tilde{f}^* \leq \left(1 - \frac{1}{4e^4 \max(kR_\infty, 1)}\right) (\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) + e^2 \delta_a + \frac{3\delta}{2}$$

where $R_\infty \leq B + \ln\left(\frac{n(\|\mathbf{A}\|_1 - \mu)}{\mu\varepsilon} + 2n\right)$ is the ℓ^∞ -radius of the sublevel set $\{(\mathbf{x}, \mathbf{y}) : \tilde{f}(\mathbf{x}, \mathbf{y}) \leq \tilde{f}(\mathbf{0}, \mathbf{0})\}$ about $(\mathbf{0}, \mathbf{0})$, where the upper bound follows from Lemma 15.2.1.

15. Quantum box-constrained Newton methods

From here on, we write $M = 4e^4 \max(kR_\infty, 1)$. The choice of δ_α and δ in the algorithm is such that $e^2\delta_\alpha + 3\delta/2 \leq \frac{\varepsilon\|\mathbf{A}(\mathbf{x})\|_1}{2M}$, hence we can also bound the progress by

$$\begin{aligned} \tilde{f}(\mathbf{x}^{(i+1)}) - \tilde{f}^* &\leq \left(1 - \frac{1}{M}\right) (\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) + \frac{\varepsilon f(\mathbf{x}^{(i)})}{2M} \\ &= \left(1 - \frac{1}{M}\right) (\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) + \frac{\varepsilon(f(\mathbf{x}^{(i)}) - \tilde{f}^*)}{2M} + \frac{\varepsilon\tilde{f}^*}{2M} \\ &\leq \left(1 - \frac{2-\varepsilon}{2M}\right) (\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) + \frac{\varepsilon\tilde{f}^*}{2M} \\ &\leq \left(1 - \frac{1}{2M}\right) (\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) + \frac{\varepsilon\tilde{f}^*}{2M} \end{aligned}$$

We show that $\tilde{f}(\mathbf{x}^{(T)}) - \tilde{f}^* \leq \varepsilon\tilde{f}^*$ for our choice of T . Note that we have

$$\begin{aligned} \tilde{f}(\mathbf{x}^{(T)}) - \tilde{f}^* &\leq \left(1 - \frac{1}{2M}\right)^T (\tilde{f}(\mathbf{0}) - \tilde{f}^*) + \sum_{i=0}^{T-1} \left(1 - \frac{1}{2M}\right)^{T-i-1} \cdot \frac{\varepsilon\tilde{f}^*}{2M} \\ &\leq \left(1 - \frac{1}{2M}\right)^T (\tilde{f}(\mathbf{0}) - \tilde{f}^*) + \left(1 - \left(1 - \frac{1}{2M}\right)^T\right) \cdot \varepsilon\tilde{f}^* \\ &\leq \left(1 - \frac{1}{2M}\right)^T (f(\mathbf{0}) - f^* + \frac{2\mu\varepsilon}{e^B}) + \varepsilon\tilde{f}^* \\ &\leq \left(1 - \frac{1}{2M}\right)^T \left(1 - \mu + \frac{2\mu\varepsilon}{e^B}\right) + \varepsilon\tilde{f}^* \\ &\leq 2\varepsilon\tilde{f}^* \end{aligned}$$

where in the third inequality we use $\tilde{f}(\mathbf{0}) = f(\mathbf{0}) + 2\mu\varepsilon/e^B$ and $f^* \leq \tilde{f}^*$, in the fourth inequality we use the potential gap bound (Lemma 14.4.1), and in the last inequality we use

$$\begin{aligned} T &= \left\lceil 8e^4 \max\left(k\left(B + \ln\left(\frac{n(1-\mu)}{\mu\varepsilon} + 2n\right)\right), 1\right) \ln\left(\frac{1-\mu+2\mu\varepsilon/e^B}{\mu\varepsilon}\right) \right\rceil \\ &\geq \left\lceil 2M \ln\left(\frac{1-\mu+2\mu\varepsilon/e^B}{\mu\varepsilon}\right) \right\rceil \\ &\geq \frac{1}{\ln\left(1 - \frac{1}{2M}\right)} \cdot \ln\left(\frac{\mu\varepsilon}{1-\mu + \frac{2\mu\varepsilon}{e^B}}\right), \end{aligned}$$

and again the inequality $\mu \leq f^* \leq \tilde{f}^*$ (Lemma 14.4.1).

This implies that

$$f(\mathbf{x}^{(T)}) - f^* \leq \tilde{f}(\mathbf{x}^{(T)}) - \tilde{f}^* + \tilde{f}^* - f^* \leq 2\varepsilon\tilde{f}^* + 3\varepsilon f^* \leq 2\varepsilon(f^* + 3\varepsilon f^*) + 3\varepsilon f^* = (5 + 6\varepsilon)\varepsilon f^*,$$

where we crucially use the last point of Lemma 15.3.1 twice, which is justified by the assumption that there exists \mathbf{x}_ε with $\|\mathbf{x}_\varepsilon\|_\infty \leq B$ with $f(\mathbf{x}_\varepsilon) \leq (1 + \varepsilon)f^*$.

Before bounding the time complexity of Algorithm 15.1, we show that throughout the algorithm, $\|\mathbf{A}(\mathbf{x}^{(i)})\|_1 \leq \tilde{f}(\mathbf{0}) \leq 1 + 2\varepsilon/e^B$ for any $i \in [T]$ with $\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^* > \varepsilon$.

Note that $\|\mathbf{A}(\mathbf{x})\|_1 \leq \tilde{f}(\mathbf{x})$ for any \mathbf{x} . Since in every iteration, $\mathbf{H}_m, \mathbf{H}_\alpha$ and \mathbf{b} satisfies the requirements of Theorem 15.1.5, we know that

$$\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}(\mathbf{x}^{(i+1)}) \geq \frac{1}{M}(\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^*) - \frac{\varepsilon}{2M},$$

where the right hand side is nonnegative if $\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^* \geq \varepsilon/2$. In other words, as long as $\tilde{f}(\mathbf{x}^{(i)}) - \tilde{f}^* > \varepsilon$, $\tilde{f}(\mathbf{x}^{(i)}) > \tilde{f}(\mathbf{x}^{(i+1)})$ and we have $\|\mathbf{A}(\mathbf{x}^{(i+1)})\|_1 \leq \tilde{f}(\mathbf{x}^{(i+1)}) \leq \tilde{f}(\mathbf{0})$.

Now, in each of the T iterations we compute:

- (i) a $C' \geq 0$ such that $C' \leq \|\mathbf{A}(\mathbf{x})\|_1 \leq 2C'$ using Lemma 15.3.2 in time $\tilde{O}(\sqrt{m})$,
- (ii) $\mathbf{H}_m, \mathbf{H}_\alpha$ as in Theorem 15.3.5 with $\delta_\alpha = \varepsilon' C' / 2e^2$ of Hess $\tilde{f}(\mathbf{x}^{(i)})$ in time $\tilde{O}(\sqrt{mn} \text{polylog}(1/\varepsilon))$ (using that $\|\mathbf{A}(\mathbf{x})\|_1 \leq C = 1 + 2\varepsilon/e^B$, $1/\mu$ are at most poly(n)),
- (iii) an $\varepsilon' \|\mathbf{A}(\mathbf{x})\|_1 / 3\ell^1$ -approximation of $\nabla \tilde{f}(\mathbf{x}^{(i)})$ in time $\tilde{O}(\sqrt{mn/\varepsilon'}) = \tilde{O}(\sqrt{Bmn/\varepsilon})$ (Lemma 15.3.6),
- (iv) an update Δ in time $\tilde{O}(n)$ using one call to the $k = O(\log(n))$ -oracle on SDD-matrices with $\tilde{O}(n)$ non-zero entries (Theorem 15.1.4).

Note that the second contribution dominates the others, resulting in an overall time complexity $\tilde{O}(B^{1.5} \sqrt{mn/\varepsilon})$. \square

Combining the above with Corollary 14.4.3 yields the following result.

Corollary 15.3.8. *For asymptotically-balanceable matrices $\mathbf{A} \in \mathbb{R}_{>0}^{n \times n}$ with m non-zero entries, one can find a $O(\varepsilon)$ - ℓ^2 -balancing vector \mathbf{x} of \mathbf{A} in time $\tilde{O}(R_\infty^{1.5} \sqrt{mn/\varepsilon})$, where R_∞ is such that there exists $\mathbf{x} \in \mathbb{R}^n$ with $f(\mathbf{x}) \leq (1 + \varepsilon^2)f^*$ and*

$$R_\infty = \|\mathbf{x}_\varepsilon\|_\infty + \ln(2n + (n(1 - \mu)/\mu\varepsilon)).$$

Unfortunately, a potential gap $f(\mathbf{x}) - f^* \leq \varepsilon f^*$ only guarantees that the rescaled matrix $\mathbf{A}' = \mathbf{A}(\mathbf{x})$ satisfies $\max_i (\sqrt{r_i(\mathbf{A}')} - \sqrt{c_i(\mathbf{A}')})^2 \leq \varepsilon$. Therefore, the squared Hellinger distance $H^2(\mathbf{r}(\mathbf{A}'), \mathbf{c}(\mathbf{A}'))$ could be $n\varepsilon$ in the worst case, with an associated quantum time complexity of $\tilde{O}(R_\infty^{1.5} \sqrt{mn/\varepsilon})$. For comparison, Osborne's algorithm yields an ε - H^2 -balancing of \mathbf{A} in quantum time $\tilde{O}(\sqrt{mn}/\varepsilon^{1.5})$ (see Theorem 14.4.6). However, for achieving closeness in ℓ^2 -distance, if R_∞ is not too large, the second-order method outperforms the classical weighted randomized variant of Osborne [ORY17], which takes time $\tilde{O}(m + n/\varepsilon^2)$, and our quantum variant Osborne's algorithm: to obtain an ε - ℓ^2 -balancing from the latter takes time $\tilde{O}(\sqrt{mn}/\varepsilon^3)$, using Lemma 13.2.4 and that the ℓ^2 -norm is smaller than the ℓ^1 -norm.

Similar to matrix scaling, in general we do not have good (i.e., polylogarithmic) bounds on the parameter R_∞ . For the general case, R_∞ is $\tilde{O}(n)$. The bound can be improved if the underlying graph G whose (weighted) adjacency matrix is \mathbf{A} is strongly connected: If the diameter of G is d , then R_∞ is $\tilde{O}(d)$ [CMTV17, Lemma 4.24]. For entrywise-positive \mathbf{A} , the diameter is 1, hence we obtain the following corollary.

Corollary 15.3.9. *For entrywise-positive matrices \mathbf{A} , one can find an ε - ℓ^2 -balancing of \mathbf{A} in time $\tilde{O}(n^{1.5}/\varepsilon)$.*

16. Quantum query lower bounds: constant precision

In this chapter we will prove an $\Omega(\sqrt{mn})$ lower bound on the query complexity of quantum algorithms for $\Theta(1)$ - ℓ^1 -scaling $n \times n$ matrices with m non-zero entries to the uniform marginals $(\mathbf{1}/n, \mathbf{1}/n)$, as well as for $\Theta(1)$ - ℓ^1 -balancing. We will do so by showing an $\Omega(n\sqrt{s})$ lower bound on instances with s potentially non-zero entries per row and column (note that with $m = ns$, we have $n\sqrt{s} = \sqrt{mn}$). The lower bounds for both problems are shown by a reduction to (partially) learning a string hidden in a permutation, a problem we define and prove a query lower bound for in Section 16.1. The lower bound for matrix scaling is proven in Section 16.2, and the lower bound for matrix balancing is shown in Section 16.3.

16.1. Partially learning a string hidden in a permutation

We first prove a query lower bound for the problem of learning a string hidden in a permutation matrix. Here we are given query access to a matrix $\mathbf{P}_{\sigma,z}$ which is an $n \times n$ permutation matrix corresponding to a permutation σ , except that the 1 entries have been replaced with the entries of some vector $z \in \{-1, 1\}^n$. In particular, denoting \mathbf{P}_σ for the permutation matrix corresponding to σ , $\mathbf{P}_{\sigma,z} = \mathbf{P}_\sigma \text{diag}(z)$ so that the i th column of $\mathbf{P}_{\sigma,z}$ contains z_i in the $\sigma(i)$ th row, i.e., $(\mathbf{P}_{\sigma,z})_{\sigma(i)i} = z_i$. Informally, the goal is to recover a constant fraction (close to 1) of the entries of z correctly.

As we will show, this problem requires $\Omega(n\sqrt{n})$ quantum queries to the entries of a dense matrix for $\mathbf{P}_{\sigma,z}$ to solve, or $\Omega(n\sqrt{s})$ quantum queries to an s -sparse input. We follow a similar proof structure as in the $\Omega(\sqrt{n})$ -query lower bound given in [Amb02] for finding $\sigma^{-1}(1)$ for a permutation $\sigma \in S_n$, and the $\Omega(n\sqrt{n})$ -query lower bound for graph connectivity given in [DHHM06]. We first prove the lower bound for fully recovering z , and then show that recovering a constant fraction of the elements of z is just as hard.

To obtain bounds for the s -sparse setting, we limit our permutations to products of n/s permutations on s elements. More precisely, if n is a multiple of s , then we shall work with an n/s -tuple $\sigma = (\sigma^1, \dots, \sigma^{n/s}) \in S_s^{\times n/s}$ of permutations of S_s . This may be identified with the permutation $\sigma \in S_n$ given by $\sigma(as + b) = as + \sigma^{a+1}(b)$ for $a \in \{0, \dots, n/s - 1\}$ and $b \in [s]$. Equivalently, $\sigma \in S_n$ is determined by \mathbf{P}_σ being the block-diagonal matrix consisting of the $s \times s$ blocks \mathbf{P}_{σ_j} for $j \in [n/s]$. This also means that dense access to \mathbf{P}_σ is equivalent to s -sparse access (we assume s is known to the algorithm). We now formally define the problem.

This chapter is adapted from [AGL+21].

Definition 16.1.1 (The (s, n) -string hidden in a permutation problem). Let $n, s \geq 1$ be integers such that n is a multiple of s . An instance of the (s, n) -string hidden in a permutation problem is a tuple (σ, z) where $\sigma \in S_s^{\times n/s}$ is an n/s -tuple of permutations of $[s]$, and a bit string $z \in \{-1, 1\}^n$. The input is accessible via dense queries to $\mathbf{P}_{\sigma, z} = \mathbf{P}_\sigma \text{diag}(z)$. For $\lambda \in (0, 1)$, we say that a bit string $\tilde{z} \in \{-1, 1\}^n$ is a $(1 - \lambda)$ -solution to the (s, n) -string hidden in a permutation problem if $\tilde{z}_i = z_i$ for at least $(1 - \lambda)n$ indices $i \in [n]$, and we say that \tilde{z} is a full solution if $\tilde{z} = z$.

Next, we recall the exact version of the adversary bound that we rely on:

Lemma 11.3.1 ([Amb02, Thm. 6.1]). Let $f: A \subseteq \Sigma^N \rightarrow B$ be a function of N variables, which takes values in some finite set B . Let $X, Y \subseteq A$ be two sets of inputs such that $f(x) \neq f(y)$ if $x \in X$ and $y \in Y$. Let $R \subseteq X \times Y$ be nonempty, and suppose that it satisfies:

- For every $x \in X$, there exist at least m_X different $y \in Y$ such that $(x, y) \in R$.
- For every $y \in Y$, there exist at least m_Y different $x \in X$ such that $(x, y) \in R$.

Let $\ell_{x,i}$ be the number of $y \in Y$ such that $(x, y) \in R$ and $x_i \neq y_i$, and similarly for $\ell_{y,i}$. Let $\ell_{\max} = \max_{i \in [N]} \max_{(x,y) \in R, x_i \neq y_i} \ell_{x,i} \ell_{y,i}$. Then any algorithm that computes f with success probability $\geq 2/3$ uses $\Omega\left(\sqrt{\frac{m_X m_Y}{\ell_{\max}}}\right)$ quantum queries to the input.

Proposition 16.1.2. Let $s \geq 2$ and let n be a positive multiple of s . Then any quantum algorithm which for every instance (σ, z) of the (s, n) -string hidden in a permutation problem fully recovers z with success probability at least $2/3$ makes at least $\Omega(n\sqrt{s})$ quantum queries to $\mathbf{P}_{\sigma, z}$.

Proof. For ease of notation we will assume that n is even, but the proof also holds for odd n with minimal tweaks. We will use Lemma 11.3.1 with the following choices: let $\Sigma = \{-1, 0, 1\}$, let $A \subseteq \Sigma^{n^2}$ be the set of all matrices of the form $\mathbf{P}_{\sigma, z}$ where $\sigma \in S_s^{\times n/s}$ and $z \in \{-1, 1\}^n$, let $B = \{-1, 1\}^n$, and let $f: A \rightarrow B$ be the function given by $f(\mathbf{P}_{\sigma, z}) = z$. This map is well-defined: z can be recovered from $\mathbf{P}_{\sigma, z}$ by computing its column sums. With this setup, it is clear that the adversary bound yields a lower bound on the number of quantum queries made to the entries of $\mathbf{P}_{\sigma, z}$ for computing f , i.e., recovering z . We let X be the set of all inputs (σ, z) where $|z| = n/2$, and Y be the set of all inputs (π, w) where $|w| = n/2 - 1$. Here we write $|z|$ for the number of 1's in the string z .

We define the relation $R \subseteq X \times Y$ as consisting of those $((\sigma, z), (\pi, w)) \in X \times Y$ for which there exist distinct $i, i' \in [n]$ such that

- π can be obtained from σ by swapping the function values for i, i' , i.e., $\pi = \sigma \circ (i\ i')$.
- $z_i = 1$ and $w_i = -1$, and $z_j = w_j$ for $j \in [n]$ with $j \neq i$.

Note that i, i' are automatically in the same block of s elements, as otherwise either π or σ is not in $S_s^{\times n/s}$. This operation corresponds to swapping columns of \mathbf{P}_σ to get \mathbf{P}_π . Furthermore, for any $((\sigma, z), (\pi, w)) \in R$, the tuple (i, i') is uniquely determined.

We now compute the quantities appearing in Lemma 11.3.1. For a given $(\sigma, z) \in X$, the number m_X of $y \in Y$ such that $(x, y) \in R$ is $(n/2) \cdot (s - 1) = \Omega(ns)$: we can pick

the index i among any of the $n/2$ columns of $\mathbf{P}_{\sigma,z}$ containing a 1, and then have $s - 1$ choices for i' left. Similarly, we have $m_Y = (n/2 + 1) \cdot (s - 1) = \Omega(ns)$.

We now consider the distinguishing power of a single query, that is, we compute ℓ_{\max} . Informally, the key observation is that the relation R is such that if a query to the (j, i) -entry of the matrix distinguishes two inputs, then on precisely one of the two inputs the query returns 0. The (j, i) -query returns 0 on input (σ, z) (in either X or Y) if and only if $\sigma(i) \neq j$. For a neighbor that we can distinguish with the (j, i) -query we thus need to swap the function values for i and $\sigma^{-1}(j)$, and this leaves us with at most a constant number of neighbors (with respect to R) that we can distinguish (the only remaining freedom is in changing one bit of the bit string). On the other hand, if the (j, i) -query returns a non-zero outcome on input (σ, z) , then it has $O(s)$ neighbors that we can distinguish with the (j, i) -query: we need to swap the i th column with any other column in the same block of s elements (and flip the bit associated with one of the two columns).

We now make this formal. Consider a query to the (j, i) -th element of the matrix and a pair $((\sigma, z), (\pi, w)) \in R$. We can distinguish two cases: $\sigma(i) \neq j$ and $\sigma(i) = j$.

- If $\sigma(i) \neq j$, then the (j, i) -query on (σ, z) results in a 0. All $(\pi', w') \in Y$ that can be distinguished from (σ, z) by the (j, i) -query must have $\pi'(i) = j$ (since otherwise the query returns 0), and if (π', w') is also in relation to (σ, z) this means $\pi' = \sigma \circ (i \sigma^{-1}(j))$. There are at most two such (π', w') (since π' is now determined, and w' must be obtained from z by flipping a bit in the i -th or $\sigma^{-1}(j)$ -th position) and thus $\ell_{(\sigma,z),(j,i)} \leq 2$. To compute $\ell_{(\pi,w),(j,i)}$ we now use that, by the preceding argument, $\pi(i) = j$. Then, if the (j, i) -query distinguishes (π, w) from $(\sigma', z') \in X$ with $((\sigma', z'), (\pi, w)) \in R$, we must have $\sigma'(i) \neq j$ and thus $\pi = \sigma' \circ (i i')$ for some i' distinct from i but in the same block of s elements. Since the relation also requires w to equal v except at the positions i and i' , this gives $\ell_{(\pi,w),(j,i)} \leq 4(s - 1)$.
- If $\sigma(i) = j$, then the (j, i) -query on (σ, z) results in either a 1 or -1 . Since $((\sigma, z), (\pi, w)) \in R$, this means $\pi(i) \neq j$. We can thus proceed as in the previous case but with the roles of (σ, z) and (π, w) reversed. We then obtain $\ell_{(\sigma,z),(j,i)} \in O(s)$ and $\ell_{(\pi,w),(j,i)} \in O(1)$.

In both cases we have $\ell_{(\sigma,z),(j,i)} \ell_{(\pi,w),(j,i)} \in O(s)$ and thus $\ell_{\max} \in O(s)$. Hence the number of queries made is

$$\Omega \left(\sqrt{\frac{m_X m_Y}{\ell_{\max}}} \right) = \Omega \left(\sqrt{\frac{n^2 s^2}{s}} \right) = \Omega(n\sqrt{s}). \quad \square$$

We now show that learning only a constant fraction of the entries of z correctly still requires $\Omega(n\sqrt{s})$ quantum queries.

Theorem 16.1.3. *There exists a constant $\lambda \in (0, 1)$ such that the following holds: Let $s \geq 2$ and let n be a positive multiple of s . Then any quantum algorithm which for every instance (σ, z) of the (s, n) -string hidden in a permutation problem recovers $(1 - \lambda)n$ of the entries of z with success probability at least $2/3$, makes at least $\Omega(n\sqrt{s})$ quantum queries to $\mathbf{P}_{\sigma,z}$.*

16. Quantum query lower bounds: constant precision

Proof. Let $N, S \in \mathbb{N}$ and $c_1 \in \mathbb{R}_{>0}$ be the constants from the big- Ω notation in Proposition 16.1.2, i.e., for any $n > N$ and $s > S$, at least $c_1 n \sqrt{s}$ quantum queries to an s -sparse input are required to fully recover z with success probability at least $2/3$.

Let $A, B \in \mathbb{N}$ and $c_2 \in \mathbb{R}_{>0}$ be the constants from Lemma 12.3.2 such that deterministic Grover search for multiple elements over a search space of size $a > A$, with at most $b > B$ marked elements, uses at most $c_2 \sqrt{ab}$ queries to find all marked elements with probability 1.

Let $\lambda = \min\{\frac{1}{3}, \frac{c_1^2}{4c_2^2}\}$. Let \mathcal{A} be an algorithm that uses at most T queries to an s -sparse input for the input $\mathbf{P}_{\sigma, z}$ and outputs a string \tilde{z} that (with probability $\geq 2/3$) agrees with z for $\mathbf{P}_{\sigma, z}$ on all but $\leq \lambda n$ elements. Searching for the elements where these strings do not agree can be done by Grover searching through all ns possible non-zero positions in the matrix, and marking those (j, i) where $(\mathbf{P}_{\sigma, z})_{ji} \neq 0$ and $(\mathbf{P}_{\sigma, z})_{ji} \neq \tilde{z}_i$. There are at most λn such elements, so we can find all of them using $c_2 \sqrt{ns \lambda n} = c_2 n \sqrt{\lambda s}$ queries to \mathbf{P} . We can now flip the erroneous bits in \tilde{z} to fully recover z . Therefore z can be identified exactly with failure probability at most $1/3$ using $T + c_2 n \sqrt{\lambda s}$ queries. It follows from Proposition 16.1.2 that (if $n > N$, $s > S$, $ns > A$, and $\lambda n > B$)

$$T + c_2 n \sqrt{\lambda s} \geq c_1 n \sqrt{s}.$$

For our specific choice of λ , this implies

$$\begin{aligned} T &\geq c_1 n \sqrt{s} - c_2 n \sqrt{\lambda s} \\ &\geq c_1 n \sqrt{s} - c_2 \sqrt{\frac{c_1^2}{4c_2^2}} n \sqrt{s} \\ &= c_1 n \sqrt{s} / 2. \end{aligned} \quad \square$$

16.2. Lower bound for matrix scaling

To obtain a query lower bound for matrix scaling, we will reduce the problem of learning the string hidden in a permutation to the matrix scaling problem. For an instance (σ, z) , this is achieved by replacing each non-zero entry of the permutation matrix \mathbf{P}_σ by one of two 2×2 gadget matrices, depending on the value of the bit string z associated with that column (and each 0-entry is replaced by the 2×2 all-0 matrix). These gadget matrices are chosen such that we can recover z from the row-scaling vectors \mathbf{x} of an $\Theta(1)$ - ℓ^1 -scaling to uniform marginals.

We will use the following gadget matrices:

$$\mathbf{B}_1 = \begin{bmatrix} \frac{1}{6} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{3} \end{bmatrix}, \quad \mathbf{B}_{-1} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{6} & \frac{1}{6} \end{bmatrix},$$

Lemma 16.2.1. *The matrices $\mathbf{B}_1, \mathbf{B}_{-1} \in \{\frac{1}{3}, \frac{1}{6}\}^{2 \times 2}$ are entrywise positive, with entries summing to one, and they are exactly scalable to uniform marginals. Let $i \in \{1, -1\}$, and suppose $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^2 \times \mathbb{R}^2$ are $\frac{1}{8}$ - ℓ^1 -scaling vectors for \mathbf{B}_i to the uniform marginals $(\frac{1}{2}, \frac{1}{2})$. Then $i = \text{sign}(x_1 - x_2)$ and $|x_1 - x_2| > 0.18$. Moreover, $((x_2, x_1), \mathbf{y})$ are $\frac{1}{8}$ - ℓ^1 -scaling vectors for \mathbf{B}_{-i} to uniform marginals.*

In other words, the matrices can be distinguished just by learning the row-scaling vectors, and they have the same set of possible row-scaling vectors.

Proof. Since one matrix is obtained by swapping the rows of the other, the last claim is immediate, and it suffices to prove the remaining claims for \mathbf{B}_1 . First, we note that \mathbf{B}_1 is exactly scalable, since

$$\begin{bmatrix} \frac{3}{4} & 0 \\ 0 & \frac{3}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{6} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

has uniform marginals. Now suppose that (\mathbf{x}, \mathbf{y}) is an $\frac{1}{8}$ - ℓ^1 -scaling of \mathbf{B}_1 to uniform marginals. By the requirement on the row marginals, we have

$$\left(\frac{1}{6}e^{y_1} + \frac{1}{6}e^{y_2} \right) e^{x_1} \geq \frac{1}{2} - \frac{1}{8} \quad \text{and} \quad \left(\frac{1}{3}e^{y_1} + \frac{1}{3}e^{y_2} \right) e^{x_2} \leq \frac{1}{2} + \frac{1}{8}.$$

By dividing the first inequality by the second one we get

$$\frac{1}{2} \cdot \frac{e^{x_1}}{e^{x_2}} \geq \frac{3}{5},$$

and so

$$x_1 - x_2 \geq \ln \frac{6}{5} > 0.18. \quad \square$$

We now prove the query lower bound for matrix scaling. This lower bound also holds if we allow the algorithm to know the set of vectors \mathbf{y} that can occur in an ε - ℓ^1 -scaling (\mathbf{x}, \mathbf{y}) of the matrix (in fact, $\mathbf{y} = 0$ is always a solution).

Theorem 16.2.2. *There exists a constant $\varepsilon \in (0, 1)$ such that any quantum algorithm which, given s -sparse access to an $n \times n$ -matrix which is exactly scalable to uniform marginals and with entries summing to one, returns an ε - ℓ^1 -scaling with probability $\geq 2/3$, requires $\Omega(n\sqrt{s})$ quantum queries.*

Proof. Without loss of generality we may assume s is a positive multiple of 8 and n is a multiple of s . Let $\varepsilon = \frac{\lambda}{16} \in \Theta(1)$, where λ is the constant from Theorem 16.1.3. Assume there is a T-query quantum algorithm \mathcal{A} that solves the ε - ℓ^1 -scaling problem with success probability $\geq 2/3$. We will construct an algorithm for recovering a $1 - \lambda$ fraction of the a string hidden in a permutation in $S_{n/2}$, given $s/2$ -sparse query access to the corresponding $n/2 \times n/2$ string-hiding permutation matrix $\mathbf{P} \in \{-1, 0, 1\}^{n/2 \times n/2}$.

Let $\mathbf{Q} \in \mathbb{R}^{n \times n}$ be a $n/2 \times n/2$ block-matrix with the following 2×2 blocks:

- If $P_{ij} = 0$ then the (i, j) -block in \mathbf{Q} is the 2×2 all-0 matrix.
- If $P_{ij} \in \{1, -1\}$ then the (i, j) -block in \mathbf{Q} is the matrix $\mathbf{B}_{P_{ij}}$ from Lemma 16.2.1.

Finally, let $\mathbf{R} = \frac{2}{n}\mathbf{Q}$. Then \mathbf{R} is an $n \times n$ matrix that has entries summing to one, and it is exactly scalable to uniform marginals. Note that s -sparse query access for \mathbf{R} can be constructed using $O(1)$ queries to an $s/2$ -sparse input for \mathbf{P} .

By applying \mathcal{A} to \mathbf{R} we obtain ε - ℓ^1 -scaling vectors (\mathbf{x}, \mathbf{y}) . Let $\mathbf{x}^{(i)} \in \mathbb{R}^2$ (resp. $\mathbf{y}^{(j)}$) denote the restriction of \mathbf{x} (resp. \mathbf{y}) to the two coordinates corresponding to the

16. Quantum query lower bounds: constant precision

i -th row (resp. the j -th column) of \mathbf{P} . We can compute the ℓ^1 -error of the row and column marginals in terms of the 2×2 -blocks $\frac{2}{n}\mathbf{B}_{P_{ij}}$ corresponding to $P_{ij} \neq 0$. Accordingly, we obtain

$$\sum_{(i,j):P_{ij} \neq 0} \left\| \mathbf{r} \left(e^{\mathbf{x}^{(i)}} \frac{2}{n} \mathbf{B}_{P_{ij}} e^{\mathbf{y}^{(j)}} \right) - \frac{\mathbf{1}}{n} \right\|_1 \leq \varepsilon \text{ and } \sum_{(i,j):P_{ij} \neq 0} \left\| \mathbf{c} \left(e^{\mathbf{x}^{(i)}} \frac{2}{n} \mathbf{B}_{P_{ij}} e^{\mathbf{y}^{(j)}} \right) - \frac{\mathbf{1}}{n} \right\|_1 \leq \varepsilon,$$

and hence

$$\sum_{(i,j):P_{ij} \neq 1} \left\| \mathbf{r}(e^{\mathbf{x}^{(i)}} \mathbf{B}_{P_{ij}} e^{\mathbf{y}^{(j)}}) - \frac{\mathbf{1}}{2} \right\|_1 + \left\| \mathbf{c}(e^{\mathbf{x}^{(i)}} \mathbf{B}_{P_{ij}} e^{\mathbf{y}^{(j)}}) - \frac{\mathbf{1}}{2} \right\|_1 \leq \varepsilon n.$$

This implies that for all but $\frac{\varepsilon n}{1/8} = \lambda n/2$ of these blocks, the corresponding $(\mathbf{x}^{(i)}, \mathbf{y}^{(j)})$ are $\frac{1}{8}$ - ℓ^1 -scaling vectors for $\mathbf{B}_{P_{ij}}$. By Lemma 16.2.1, we can therefore correctly identify P_{ij} from $\mathbf{x}^{(i)}$ for $(1 - \lambda)n/2$ rows, and hence we learn the bits of the hidden string for those i .

Since this identifies $(1 - \lambda)n/2$ of the $n/2$ hidden bits, and because the algorithm has failure probability at most $1/3$, it follows from Theorem 16.1.3 that the number of queries T is $\Omega(n\sqrt{s})$. \square

Corollary 16.2.3. *There exists a constant $\varepsilon \in (0, 1)$ such that any quantum algorithm which, given sparse query access an $n \times n$ -matrix that is exactly scalable to uniform marginals and has m potentially non-zero entries which sum to 1, returns an ε - ℓ^1 -scaling with probability $\geq 2/3$, requires $\Omega(\sqrt{mn})$ quantum queries to the matrix.*

16.3. Lower bound for matrix balancing

To obtain a query lower bound for matrix balancing we will follow the same strategy as in Section 16.2. We consider an instance of the (s, n) -bit string hidden in a permutation problem as in Definition 16.1.1, and create from it a matrix $3n \times 3n$ -matrix \mathbf{B} with $\Theta(ns)$ possibly non-zero entries, such that if $\mathbf{B}(\mathbf{x})$ is ε - ℓ^1 -balanced one can recover a large constant fraction of the bits z_j from \mathbf{x} . The matrix \mathbf{B} is constructed using the following gadget matrices:

$$\mathbf{B}_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 2 \\ 2 & 1 & 0 \end{bmatrix}, \quad \mathbf{B}_{-1} = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 1 \\ 1 & 2 & 0 \end{bmatrix}. \quad (16.3.1)$$

Lemma 16.3.1. *The matrices $\mathbf{B}_1, \mathbf{B}_{-1} \in \{0, 1, 2\}^{3 \times 3}$ are exactly balanceable. For $i \in \{-1, 1\}$, let $\mathbf{x} \in \mathbb{R}^3$ be such that $\mathbf{B}_i(\mathbf{x})$ is ε - ℓ^∞ -balanced for $\varepsilon = 1/100$. Then $i = \text{sign}(x_1 - x_2)$ and $|x_1 - x_2| \geq \frac{1}{2}$.*

Proof. We start by showing that these matrices are exactly balanceable. For \mathbf{B}_1 , let $x_1 = \ln(\sqrt{2})$, $x_2 = -\ln(\sqrt{2})$ and $x_3 = 0$, Then

$$e^{\text{diag}(\mathbf{x})} \mathbf{B}_1 e^{-\text{diag}(\mathbf{x})} = \begin{bmatrix} 0 & 0 & e^{x_1 - x_3} \\ 0 & 0 & 2e^{x_2 - x_3} \\ 2e^{x_3 - x_1} & e^{x_3 - x_2} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & \sqrt{2} \\ 0 & 0 & \sqrt{2} \\ \sqrt{2} & \sqrt{2} & 0 \end{bmatrix}, \quad (16.3.2)$$

and similar for \mathbf{B}_{-1} .

Now let \mathbf{x} be such that $\mathbf{B}_1(\mathbf{x})$ is ε - ℓ^∞ -balanced, for some ε that we pick later. Let $a = e^{x_3 - x_1}$ and $b = e^{x_2 - x_3}$, so

$$\mathbf{B}_1(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 1/a \\ 0 & 0 & 2b \\ 2a & 1/b & 0 \end{bmatrix}.$$

We are interested in giving a lower bound on $\ln(1/(ab))$, as this equals $x_1 - x_2$. To do so we first bound $\|\mathbf{B}_1(\mathbf{x})\|_1$ from above. Observe first that

$$\|\mathbf{B}_1(\mathbf{x})\|_1 = 2a + 2b + \frac{1}{a} + \frac{1}{b} \leq 2(\max\{2a, 1/a\} + \max\{2b, 1/b\}).$$

If $\mathbf{B}_1(\mathbf{x})$ is ε - ℓ^∞ -balanced, then in particular we must have

$$\begin{aligned} \max\{2a, 1/a\} - \sqrt{2} &\leq \left| 2a - \frac{1}{a} \right| \leq \varepsilon \|\mathbf{B}_1(\mathbf{x})\|_1 \leq 2\varepsilon(\max\{2a, 1/a\} + \max\{2b, 1/b\}), \\ \max\{2b, 1/b\} - \sqrt{2} &\leq \left| 2b - \frac{1}{b} \right| \leq \varepsilon \|\mathbf{B}_1(\mathbf{x})\|_1 \leq 2\varepsilon(\max\{2a, 1/a\} + \max\{2b, 1/b\}). \end{aligned}$$

since $|2a - \frac{1}{a}| = |c_1(\mathbf{B}_1(\mathbf{x})) - r_1(\mathbf{B}_1(\mathbf{x}))|$ and similar for b . Note that the bound $\max\{2a, 1/a\} - \sqrt{2} \leq |2a - 1/a|$ follows from a simple case distinction: if $a \geq 1/\sqrt{2}$ then $0 \leq \max\{2a, 1/a\} - \sqrt{2} = 2a - \sqrt{2} \leq 2a - 1/a$, and a similar argument if $a \leq 1/\sqrt{2}$. This gives

$$\begin{aligned} \max\{2a, 1/a\} &\leq \frac{2\varepsilon}{1-2\varepsilon} \max\{2b, 1/b\} + \frac{\sqrt{2}}{1-2\varepsilon}, \\ \max\{2b, 1/b\} &\leq \frac{2\varepsilon}{1-2\varepsilon} \max\{2a, 1/a\} + \frac{\sqrt{2}}{1-2\varepsilon}. \end{aligned}$$

Therefore

$$\left(1 - \frac{4\varepsilon^2}{(1-2\varepsilon)^2}\right) \max\{2a, 1/a\} \leq \frac{2\sqrt{2}\varepsilon}{(1-2\varepsilon)^2} + \frac{\sqrt{2}}{1-2\varepsilon},$$

which simplifies to

$$\frac{1-4\varepsilon}{(1-2\varepsilon)^2} \max\{2a, 1/a\} \leq \frac{2\sqrt{2}\varepsilon + \sqrt{2}(1-2\varepsilon)}{(1-2\varepsilon)^2} = \frac{\sqrt{2}}{(1-2\varepsilon)^2}$$

so $\max\{2a, 1/a\} \leq \sqrt{2}/(1-4\varepsilon)$ which is upper bounded by $2\sqrt{2}$ when $\varepsilon < 1/8$. The same bound holds on $\max\{2b, 1/b\}$, so $\|\mathbf{B}_1(\mathbf{x})\|_1 \leq 8\sqrt{2}$ when $\varepsilon < 1/8$.

Now we are ready to lower bound $\ln(1/(ab)) = x_1 - x_2$. If $\mathbf{B}_1(\mathbf{x})$ is ε - ℓ^∞ -balanced for $\varepsilon < 1/8$, then $|c_j(\mathbf{B}_1(\mathbf{x})) - r_j(\mathbf{B}_1(\mathbf{x}))| \leq \varepsilon \|\mathbf{B}_1(\mathbf{x})\|$ for $j = 1, 2$ yields

$$\begin{aligned} 2a &\leq 1/a + \varepsilon \|\mathbf{B}_1(\mathbf{x})\|_1 \leq 1/a + 8\sqrt{2}\varepsilon, \\ 2b &\leq 1/b + \varepsilon \|\mathbf{B}_1(\mathbf{x})\|_1 \leq 1/b + 8\sqrt{2}\varepsilon, \end{aligned}$$

16. Quantum query lower bounds: constant precision

where we used the previously established bound $\|\mathbf{B}_1(\mathbf{x})\|_1 \leq 8\sqrt{2}$. Multiplying the first inequality by a , the second by b , and solving both in terms of these factors gives

$$\begin{aligned} a &\leq 2\sqrt{2}\varepsilon + \sqrt{\frac{1}{2} + 8\varepsilon^2}, \\ b &\leq 2\sqrt{2}\varepsilon + \sqrt{\frac{1}{2} + 8\varepsilon^2}, \end{aligned}$$

and hence

$$\frac{1}{ab} \geq \frac{1}{(2\sqrt{2}\varepsilon + \sqrt{\frac{1}{2} + 8\varepsilon^2})^2}$$

Note that the right hand side increases as ε decreases. It can easily be verified that for $\varepsilon = 1/100$ the right hand side is larger than $\sqrt{\varepsilon}$ (and hence for all smaller ε as well), and hence

$$x_1 - x_2 \geq \frac{1}{2}.$$

With a similar argument we find that for \mathbf{B}_{-1} we have $x_1 - x_2 \leq -\frac{1}{2}$. \square

Before we proceed with the main lower bound argument, we prove a simple inequality which we later use to relate ε - ℓ^1 -balancings of a block-diagonal matrix to the quality of the balancing of the individual blocks. This is more complicated than in the matrix scaling setting because the definition of \mathbf{B} being ε - ℓ^1 -balanced is $\|\mathbf{r}(\mathbf{B}) - \mathbf{c}(\mathbf{B})\|_1 \leq \varepsilon\|\mathbf{B}\|_1$, which does not imply the same bound for individual blocks; indeed, one of the blocks could be very unbalanced and have large 1-norm, which improves the relative quality of the other blocks.

Lemma 16.3.2. *For any $\mathbf{x} \in \mathbb{R}^n$, the matrices \mathbf{B}_i defined in Eq. (16.3.1) satisfy*

$$\|\mathbf{r}(\mathbf{B}_i(\mathbf{x})) - \mathbf{c}(\mathbf{B}_i(\mathbf{x}))\|_1 \geq \|\mathbf{B}_i(\mathbf{x})\|_1 - 4\sqrt{2}.$$

In fact, the bound is true with the right-hand side multiplied by a factor 2, but the argument is more complicated. Note that this bound is also tight for the choice $\mathbf{x} = \mathbf{x}^*$, and that we do **not** assume that $\mathbf{B}_i(\mathbf{x})$ is ε -balanced for any $\varepsilon > 0$.

Proof. Clearly it suffices to show the inequality for \mathbf{B}_1 . We first observe that for any $c > 0$, one has

$$\left|2c - \frac{1}{c}\right| - \left(2c + \frac{1}{c}\right) \geq -2\sqrt{2}.$$

Indeed, if $c \geq \frac{1}{\sqrt{2}}$, then

$$\left|2c - \frac{1}{c}\right| - \left(2c + \frac{1}{c}\right) = 2c - \frac{1}{c} - \left(2c + \frac{1}{c}\right) = -\frac{2}{c} \geq -2\sqrt{2} \quad (16.3.3)$$

where the last inequality holds because $c \geq \frac{1}{\sqrt{2}}$. The case for $c \leq \frac{1}{\sqrt{2}}$ is similar. From this inequality it follows that

$$\|\mathbf{r}(\mathbf{B}_1(\mathbf{x})) - \mathbf{c}(\mathbf{B}_1(\mathbf{x}))\| - \|\mathbf{B}_1(\mathbf{x})\|_1$$

$$\begin{aligned}
 &= |e^{x_1-x_3} - 2e^{x_3-x_1}| + |2e^{x_2-x_3} - e^{x_3-x_2}| + |r_3(\mathbf{B}_1(\mathbf{x})) - c_3(\mathbf{B}_1(\mathbf{x}))| \\
 &\quad - (e^{x_1-x_3} + 2e^{x_3-x_1} + 2e^{x_2-x_3} + e^{x_3-x_2}) \\
 &\geq -4\sqrt{2}
 \end{aligned}$$

where we applied Eq. (16.3.3) with $c = e^{x_3-x_1}$ and $c = e^{x_2-x_3}$, and $|r_3(\mathbf{B}_1(\mathbf{x})) - c_3(\mathbf{B}_1(\mathbf{x}))| \geq 0$. \square

Theorem 16.3.3. *Let $s > 2$ and let $n \geq s$. There exists a constant $\varepsilon \in (0, 1)$ such that any quantum algorithm which, given s -sparse query access to an $n \times n$ -matrix with entries summing to one that can be balanced exactly, returns an ε - ℓ^1 -balancing with probability $\geq 2/3$, requires $\Omega(n\sqrt{s})$ quantum queries to the input.*

Proof. Without loss of generality we may assume that n is a multiple of s ; if this is not the case, then we may round down n to a multiple n' of s , prove the lower bound for n' , and direct sum the hard $n' \times n'$ instances with a sparse exactly balanced matrix with sufficiently small entries to obtain a hard $n \times n$ instance. Note that the block must have sufficiently small entries because their size affects the relative quality of the hard instance.

We now show how to reduce the problem of finding a (s, n) -string hidden in a permutation (Definition 16.1.1) to a balancing instance. Let $\sigma \in S_s^{n/s}$ and $z \in \{-1, 1\}^n$, and let \mathbf{P}_σ be the permutation matrix of σ when viewed as a permutation of $[n]$. We define a $3n \times 3n$ -matrix \mathbf{B} as follows. We start from the matrix $\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{P}_\sigma^\top \\ \mathbf{P}_\sigma & \mathbf{0} \end{bmatrix}$. For every $j \in [n]$, if $z_j = 1$, we replace the $(j, \sigma(j) + n)$ -th

entry of \mathbf{A} by the vector $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$, and the $(\sigma(j) + n, j)$ -th entry by the vector $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$.

If $z_j = -1$, we replace the $(j, \sigma(j) + n)$ -th entry of \mathbf{A} by the vector $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$, and the $(\sigma(j) + n, j)$ -th entry by the vector $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$. Lastly, we replace zero entries by a zero matrix of an appropriate size, depending on whether it is in the top-left block (2×2), top-right (2×1), bottom-left (1×2) or bottom-right (1×1). In other words, we have constructed \mathbf{B} so that the 3×3 submatrix given by selecting the $2j - 1, 2j$ and $\sigma(j) + n$ 'th row and $2j - 1, 2j$ and $\sigma(j) + n$ 'th column is exactly $\mathbf{B}_{z_j}^j$, and the rest of the entries are 0. We shall refer to this submatrix as \mathbf{B}^j from here onwards. Furthermore, it is clear that one can implement query access to \mathbf{B} using a constant number of queries to $\mathbf{P}_{\sigma, z}$. We now show that from an appropriate rebalancing of \mathbf{B} , one can recover the z_j 's. Let $\mathbf{x} \in \mathbb{R}^{3n}$ be such that $\mathbf{B}(\mathbf{x})$ is ε - ℓ^1 -balanced, for some $\varepsilon > 0$ that we choose later. In other words, we have

$$\|\mathbf{r}(\mathbf{B}(\mathbf{x})) - \mathbf{c}(\mathbf{B}(\mathbf{x}))\|_1 \leq \varepsilon \|\mathbf{B}(\mathbf{x})\|_1.$$

The left- and right-hand side split as the sum of the 1-norms of each of the blocks, hence for $\mathbf{x}|_j = (x_{2j-1}, x_{2j}, x_{\sigma(j)+n})$ we have

$$\sum_{j=1}^n \|\mathbf{r}(\mathbf{B}^j(\mathbf{x}|_j)) - \mathbf{c}(\mathbf{B}^j(\mathbf{x}|_j))\|_1 - \varepsilon \|\mathbf{B}^j(\mathbf{x}|_j)\|_1 \leq 0.$$

We now turn this into a statement about most indices $j \in [n]$. Consider a sum $\sum_{j=1}^n s_j \leq 0$ where $s_j \geq -c$ for some $c \geq 0$. Then for a given $a > 0$, at

16. Quantum query lower bounds: constant precision

most $k_a = nc/(a + c)$ j 's can satisfy $s_j \geq a$, for otherwise $\sum_{j=1}^n s_j > ak_a - (n - k_a)c = k_a(a + c) - nc = 0$ would be positive. We can apply this with $c = 4\sqrt{2}\varepsilon$ by Lemma 16.3.2, and with $a = c(1 - \lambda)/\lambda$ where $\lambda \in (0, 1)$ is as in Theorem 16.1.3. This gives $k_a = nc/(a + c) \leq \lambda n$, so $s_j = \|\mathbf{r}(\mathbf{B}^j(\mathbf{x}_{|j})) - \mathbf{c}(\mathbf{B}^j(\mathbf{x}_{|j}))\|_1 - \varepsilon \|\mathbf{B}^j(\mathbf{x}_{|j})\|_1 \leq a$ for at least $(1 - \lambda)n$ j 's. Since $a = c(1 - \lambda)/\lambda$, we get $\|\mathbf{r}(\mathbf{B}^j(\mathbf{x}_{|j})) - \mathbf{c}(\mathbf{B}^j(\mathbf{x}_{|j}))\|_1 - \varepsilon \|\mathbf{B}^j(\mathbf{x}_{|j})\|_1 \leq 4\sqrt{2}\varepsilon(1 - \lambda)/\lambda$, hence $\|\mathbf{r}(\mathbf{B}^j(\mathbf{x}_{|j})) - \mathbf{c}(\mathbf{B}^j(\mathbf{x}_{|j}))\|_1 \leq 4\sqrt{2}\varepsilon(1 - \lambda)/\lambda + \varepsilon \|\mathbf{B}^j(\mathbf{x}_{|j})\|_1$. Now we use that $\|\mathbf{B}^j(\mathbf{x}_{|j})\|_1 \geq 4\sqrt{2}$, which holds because the exact balancing has norm $4\sqrt{2}$ (Eq. (16.3.2)), and the (logarithm of) the 1-norm forms a convex potential for the balancing problem (Section 15.3). Therefore we obtain

$$\|\mathbf{r}(\mathbf{B}^j(\mathbf{x}_{|j})) - \mathbf{c}(\mathbf{B}^j(\mathbf{x}_{|j}))\|_1 \leq \frac{\varepsilon}{\lambda} \|\mathbf{B}^j(\mathbf{x}_{|j})\|_1 \quad (16.3.4)$$

for at least $(1 - \lambda)n$ of the j 's. For $\varepsilon = \frac{\lambda}{300}$, Eq. (16.3.4) implies that for every such j , $\mathbf{B}^j(\mathbf{x}_{|j})$ is $1/300$ - ℓ^1 -balanced, hence $1/100$ - ℓ^∞ -balanced, and we can recover the corresponding z_j 's as $\text{sign}(x_{2j-1} - x_{2j})$ by Lemma 16.3.1. By Theorem 16.1.3, recovering a $1 - \lambda$ fraction of the z_j 's requires $\Omega(n\sqrt{s})$ queries to $\mathbf{P}_{\sigma, z}$, hence finding an ε_0 - ℓ^1 -balancing for \mathbf{B} must also use $\Omega(n\sqrt{s})$ queries (as we can implement queries \mathbf{B} with 2 queries to $\mathbf{P}_{\sigma, z}$). \square

As a consequence of Lemma 13.2.4, we also obtain a lower bound for balancing with constant squared Hellinger distance:

Corollary 16.3.4. *Let $s > 2$ and let $n \geq s$. There exists a constant $\varepsilon \in (0, 1)$ There exists a constant $\varepsilon \in (0, 1)$ such that any quantum algorithm which, given s -sparse query access to an $n \times n$ -matrix with entries summing to one that can be balanced exactly, returns an ε - H^2 -balancing with probability $\geq 2/3$, requires $\Omega(n\sqrt{s})$ quantum queries to the input.*

17. Quantum query lower bounds: high precision

In this chapter we prove three lower bounds: an $\tilde{\Omega}(m)$ -lower bound for $1/\text{poly}(n)$ - ℓ^2 -scaling $n \times n$ matrices with at most m non-zero entries, an $\Omega(n^2)$ -lower bound for $1/\text{poly}(n)$ - ℓ^2 -balancing $n \times n$ matrices, and for $\varepsilon \in [1/n, 1/2]$ an $\Omega(n^{1.5}/\sqrt{\varepsilon})$ -lower bound for ε - ℓ^1 -approximation of the row-sum vector of a normalized $n \times n$ matrix (with non-negative entries). The proofs for all the first two lower bounds are based on a reduction from the lower bound given below in Section 17.1. In Section 17.2 we construct the associated instances for matrix scaling, and in Section 17.3 we analyze their column marginals after a single iteration of the Sinkhorn algorithm. Afterwards, in Section 17.4 we show that these column marginals are close enough to the target marginals for the reduction to matrix scaling to work, and in Section 17.5 we put the ingredients together, with the main theorem being Theorem 17.5.2. In Section 17.6 we provide a lower bound for matrix balancing, Theorem 17.6.4, by creating a set of hard instances that form the analogue of the set of instances used for the matrix scaling lower bound. Finally, in Section 17.7 we prove a stronger lower bound than Theorem 17.1.1 for computing approximations to the row marginals.

17.1. The basic lower bound

The lower bound we reduce from is the following:

Theorem 17.1.1. *Let n be even, $\tau \in [1/n, 1/2]$ such that $n\tau$ is an integer, and let $k \geq 1$ be an integer. Given k binary strings $z^1, \dots, z^k \in \{\pm 1\}^n$, where z^i has Hamming weight $n/2 + a_i\tau n$ for $a_i \in \{-1, 1\}$, computing with probability $\geq \exp(-k/100)$ a string $\tilde{a} \in \{-1, 1\}^k$ that agrees with a in $\geq 99\%$ of the positions requires $\Omega(k/\tau)$ quantum queries.*

Proof. Let $\mathcal{D} = \{z \in \{\pm 1\}^n : |z| = n/2 + \tau n \text{ or } |z| = n/2 - \tau n\}$ and define the partial Boolean function $f: \mathcal{D} \rightarrow \{\pm 1\}$ as

$$f(z) = \begin{cases} 1 & \text{if } |z| = n/2 + \tau n \\ -1 & \text{if } |z| = n/2 - \tau n. \end{cases}$$

It is known that computing f with success probability at least $2/3$ takes $\Theta(1/\tau)$ quantum queries to z [NW99, Cor. 1.2], i.e., the bounded-error quantum query complexity $Q_{1/3}(f)$ is $\Theta(1/\tau)$.

We now proceed with bounding the query complexity of computing 99% of the entries of $f^{(k)}: \mathcal{D}^k \rightarrow \{\pm 1\}^k$ defined by $f^{(k)}(z^1, \dots, z^k) = (f(z^1), \dots, f(z^k))$. We will

This chapter is adapted from [GN22].

make use of the general adversary bound $\text{Adv}^\pm(f)$ [HLŠ07] which is known to satisfy $\text{Adv}^\pm(f) = \Theta(Q_{1/3}(f))$ [LMR+11, Thm. 1.1]. The strong direct product theorem of Lee and Roland [LR13, Thm. 5.4] says that for every $0 \leq \rho < 1$, $\mu \in [\frac{1+\sqrt{\rho}}{2}, 1]$ and integers k, K , every quantum algorithm that outputs a bit string $\tilde{a} \in \{\pm 1\}^k$, and makes T quantum queries to the bit strings z^1, \dots, z^k with

$$T \leq \frac{k\rho}{K(1-\rho)} \text{Adv}^\pm(f)$$

has the property that \tilde{a} agrees with $f^{(k)}(z^1, \dots, z^k)$ on at least a μ -fraction of the entries with probability at most $\exp(k(\frac{1}{K} - D(\mu \parallel \frac{1+\sqrt{\rho}}{2})))$.¹ Here $D(\mu \parallel \frac{1+\sqrt{\rho}}{2})$ is the Kullback–Leibler divergence between the distributions $(\mu, 1-\mu)$ and $(\frac{1+\sqrt{\rho}}{2}, \frac{1-\sqrt{\rho}}{2})$. For $\mu = 0.99$, $\rho = 0.1$ and $K = 3$, one has $\frac{1}{K} - D(\mu \parallel \frac{1+\sqrt{\rho}}{2}) \approx -0.03 \leq -1/100$. Therefore, the strong direct product theorem shows that computing 99% of the entries of $f^{(k)}(z^1, \dots, z^k) = a$ correctly, with success probability at least $\exp(-k/100)$, takes $\Omega(k \text{Adv}^\pm(f)) = \Omega(k Q_{1/3}(f)) = \Omega(k/\tau)$ quantum queries. \square

We will use this lower bound with $k = n/2$ and $\tau = 1/n$. The following intuition is useful to keep in mind. For a fixed $b \geq 2$, define the $2k \times n$ matrix \mathbf{A} whose $(2i-1)$ -th row equals $1 + z^i/b$ and whose $(2i)$ -th row equals $1 - z^i/b$. Then \mathbf{A} has the property that the row-marginals encode the Hamming weights of the z^i , and are all very close to n . (This implies that the first row-rescaling step of Sinkhorn’s algorithm encodes the a_i .) Moreover, the column-marginals are exactly uniform. Hence, one may hope that all sufficiently precise scalings of \mathbf{A} to uniform targets have scaling factors that are close to those given by the first row-rescaling step of Sinkhorn’s algorithm (and hence learn most of the a_i).

Below we formalize this approach. We show that if one randomly permutes the coordinates of each z^i (independently over i), then with high probability, all ε -scalings of the resulting matrix \mathbf{A}^σ are close to the first step of Sinkhorn’s algorithm; here we need to choose b sufficiently large ($\sim \sqrt{\ln(n)}$) and ε sufficiently small ($\sim \frac{1}{n^2 b}$). The section is organized as follows. In Section 17.2 we formally define our matrix scaling instances and we analyse the first row-rescaling step of Sinkhorn’s algorithm. In Section 17.3 we show that after the row-rescaling step, with high probability (over the choice of permutations), the column-marginals are close to uniform. In Sections 17.4 and 17.5 we use the strong convexity of the potential f from Eq. (13.1.3) to show that if the above event holds, then all approximate minimizers of f can be used to solve the counting problem.

17.2. Definition of the scaling instances and analysis of row marginals

Let $n \geq 4$ be even. Let $k = n/2$ and let $z^1, \dots, z^k \in \{\pm 1\}^n$ have Hamming weight $|z^i| = |\{j : z_j^i = 1\}| = n/2 + a_i$ for $a_i \in \{\pm 1\}$. Sample uniformly random permutations $\sigma^1, \dots, \sigma^k \in S_n$ and define w^i by $w_j^i = z_{(\sigma^i)^{-1}(j)}^i$. Let $b \geq 2$ be some

¹In [LR13] the upper bound on T is stated in terms of $\text{Adv}^*(F)$ where $F = (\delta_{f(x), f(y)})_{x, y \in \mathcal{D}}$ is the Gram matrix of f . For Boolean functions f one has $\text{Adv}^*(F) = \text{Adv}^\pm(f)$ [LMR+11, Thm. 3.4].

number depending on n , and consider the $2k \times n$ matrix \mathbf{A}^σ whose entries are $\mathbf{A}_{2i-1,j}^\sigma = 1 + \frac{w_j^i}{b}$ and $\mathbf{A}_{2i,j}^\sigma = 1 - \frac{w_j^i}{b}$. Then each column sum $c_j(\mathbf{A}^\sigma)$ is $2k$, and the row sums of \mathbf{A}^σ are given by

$$r_{2i-1}(\mathbf{A}^\sigma) = n + \frac{1}{b} \sum_{j=1}^n w_j^i = n + \frac{2}{b} a_i, \quad r_{2i}(\mathbf{A}^\sigma) = n - \frac{2}{b} a_i.$$

Let

$$X_{2i-1} = \frac{1}{2k} \cdot \frac{1}{n + \frac{2}{b} a_i} \quad \text{and} \quad X_{2i} = \frac{1}{2k} \cdot \frac{1}{n - \frac{2}{b} a_i} \quad \text{for all } i \in [k] \quad (17.2.1)$$

be the row scaling factors obtained from a single Sinkhorn step. We first observe that the difference between $x_{2i-1} := \ln(X_{2i-1})$ and $x_{2i} := \ln(X_{2i})$ permits to recover a_i .

Lemma 17.2.1. *For the specific row-scaling factors \mathbf{X} for \mathbf{A}^σ given in (17.2.1), for every $i \in [k]$ it holds that*

$$|\ln(X_{2i-1}/X_{2i})| \geq \frac{4}{nb},$$

and $\text{sign}(\ln(X_{2i}/X_{2i-1})) = a_i$.

Proof. Observe that ($nb > 2$ and therefore)

$$|\ln(X_{2i-1}/X_{2i})| = \left| \ln \left(\frac{n + \frac{2}{b} a_i}{n - \frac{2}{b} a_i} \right) \right| = \ln \left(\frac{nb + 2}{nb - 2} \right) \geq \frac{4}{nb}. \quad \square$$

17.3. Concentration of column marginals

We first give an explicit expression for the j th column marginal of $\mathbf{X}\mathbf{A}^\sigma$ where \mathbf{X} is given in (17.2.1).

Lemma 17.3.1. *The matrix $\mathbf{X}\mathbf{A}^\sigma$ has column sums*

$$c_j(\mathbf{X}\mathbf{A}^\sigma) = \frac{1}{2k(n^2 - 4/b^2)} \left(2kn - \frac{4}{b^2} \sum_{i=1}^k w_j^i a_i \right) \quad \text{for } j \in [n].$$

Proof. We have

$$\begin{aligned} c_j(\mathbf{X}\mathbf{A}^\sigma) &= \sum_{i=1}^k \left(\frac{1 + w_j^i/b}{2k(n + 2a_i/b)} + \frac{1 - w_j^i/b}{2k(n - 2a_i/b)} \right) \\ &= \frac{1}{2k(n^2 - 4/b^2)} \sum_{i=1}^k \left((1 + w_j^i/b)(n - 2a_i/b) + (1 - w_j^i/b)(n + 2a_i/b) \right) \\ &= \frac{1}{2k(n^2 - 4/b^2)} \sum_{i=1}^k \left(2n - \frac{4w_j^i a_i}{b^2} \right) \\ &= \frac{1}{2k(n^2 - 4/b^2)} \left(2kn - \frac{4}{b^2} \sum_{i=1}^k w_j^i a_i \right). \quad \square \end{aligned}$$

17. Quantum query lower bounds: high precision

We now show that with high probability (over the choice of permutations) the column marginals are close to uniform. To do so, we first compute the expectation of $\sum_{i=1}^k w_j^i a_i$ (Corollary 17.3.3). This quantity allows us to obtain the desired concentration of the column marginals via Hoeffding's inequality (Lemma 17.3.4).

Lemma 17.3.2. *Let $I = \{i \in [k] : a_i = 1\}$ and $I^c = [k] \setminus I$. Define random variables W_j, W_j^c by*

$$W_j = \sum_{i \in I} w_j^i, \quad W_j^c = \sum_{i \in I^c} w_j^i.$$

Then $\mathbb{E}[W_j] = \frac{2|I|}{n}$ and $\mathbb{E}[W_j^c] = -\frac{2|I^c|}{n}$.

Proof. Observe that each w_j^i is 1 with probability $\frac{1}{2} + \frac{a_i}{n}$ because σ^i is chosen uniformly randomly from S_n , and is -1 with probability $\frac{1}{2} - \frac{a_i}{n}$. Therefore $\mathbb{E}[w_j^i] = \frac{2a_i}{n}$. By linearity of expectation, the result follows. \square

Corollary 17.3.3. *We have*

$$\mathbb{E} \left[\sum_{i=1}^k w_j^i a_i \right] = \mathbb{E}[W_j] - \mathbb{E}[W_j^c] = \frac{2(|I| + |I^c|)}{n} = \frac{2k}{n}.$$

Lemma 17.3.4. *For $t \geq 0$ and $j \in [n]$, with probability at least $1 - 2e^{-t^2/2}$, we have*

$$\left| c_j(\mathbf{X}\mathbf{A}^\sigma) - \frac{1}{n} \right| = O\left(\frac{t}{b^2 n^2 \sqrt{k}}\right).$$

Proof. Observe first that

$$\begin{aligned} \left| c_j(\mathbf{X}\mathbf{A}^\sigma) - \frac{1}{n} \right| &= \left| \frac{1}{2k(n^2 - 4/b^2)} \left(2kn - \frac{4}{b^2} \sum_{i=1}^k w_j^i a_i \right) - \frac{1}{n} \right| \\ &= \frac{1}{2kn(n^2 - 4/b^2)} \left| n \left(2kn - \frac{4}{b^2} \sum_{i=1}^k w_j^i a_i \right) - 2k(n^2 - \frac{4}{b^2}) \right| \\ &= \frac{1}{2kn(n^2 - 4/b^2)} \left| \frac{8k}{b^2} - \frac{4n}{b^2} \sum_{i=1}^k w_j^i a_i \right| \\ &= \frac{4}{2kn(n^2 - 4/b^2)b^2} \left| 2k - n \sum_{i=1}^k w_j^i a_i \right|. \end{aligned}$$

For fixed j and distinct $i, i' \in [k]$, w_j^i and $w_j^{i'}$ are independently distributed random variables because σ^i and $\sigma^{i'}$ are independent. Therefore, $V_j := W_j - W_j^c = \sum_{i=1}^k w_j^i a_i$ is a sum of k independent random variables, with each $a_i w_j^i \in [-1, 1]$, and Hoeffding's inequality yields for any $t \geq 0$ that

$$\Pr[|V_j - \mathbb{E}[V_j]| \geq t \cdot \sqrt{k}] \leq 2 \exp(-t^2/2).$$

Assuming that $|V_j - \mathbb{E}[V_j]| \leq t\sqrt{k}$, we have

$$\left| 2k - n \sum_{i=1}^k a_i w_j^i \right| = n |\mathbb{E}[V_j] - V_j| \leq nt\sqrt{k}.$$

With this estimate, we see that

$$\left| c_j(\mathbf{X}\mathbf{A}^\sigma) - \frac{1}{n} \right| \leq \frac{4}{2kn(n^2 - 4/b^2)b^2} \cdot nt\sqrt{k} = \frac{2t}{b^2(n^2 - 4/b^2)\sqrt{k}}. \quad \square$$

Corollary 17.3.5. For any $t \geq 0$, with probability $\geq 1 - 2ne^{-t^2/2}$, we have

$$\left\| c(\mathbf{X}\mathbf{A}^\sigma) - \frac{\mathbf{1}}{n} \right\|_2 \leq \frac{2\sqrt{nt}}{b^2(n^2 - 4/b^2)\sqrt{k}} = O\left(\frac{t}{b^2n^2}\right).$$

17.4. Strong convexity properties of the potential

For a λ -strongly convex function f , the set $\{z : \|\nabla f(z)\|_2 \leq \varepsilon\}$ has a diameter that is bounded by a function of λ (we make this well-known fact precise in Lemma 17.4.3). We show that our potential is strongly convex when viewed as a function from (a suitable subset of) the linear subspace $V = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^n : \langle (x, y), (\mathbf{1}_n, -\mathbf{1}_n) \rangle = 0\}$ to \mathbb{R} (note that f is invariant under translation by multiples of $(\mathbf{1}_n, -\mathbf{1}_n)$). We use this to show that whenever $\|\text{grad } f(x, y)\|_2$ is small, (x, y) is close to the minimizer of f on V . It is easy to verify that Corollary 17.3.5 in fact gives an upper bound on the norm of the gradient at $(\ln(\mathbf{X}), \mathbf{0})$ (with \mathbf{X} as in (17.2.1)). This implies that $(\ln(\mathbf{X}), \mathbf{0})$ is close to the minimizer of f on V , and by the triangle inequality, is also close to any other (x, y) for which $\|\text{grad } f(x, y)\|_2$ is small. In the rest of this section we make the above precise.

In Lemma 17.4.1 we show that the Hessian of f restricted to V has smallest eigenvalue at least $n \cdot \mu(x, y)$ where $\mu(x, y)$ is the smallest entry appearing in $(A_{ij}e^{x_i+y_j})_{i,j}$. In Lemma 17.4.2 we show that $\mu(x^*, y^*) = \Theta(1/n^2)$. This implies that $\mu(x, y) = \Theta(1/n^2)$ for all (x, y) that are a constant distance away from (x^*, y^*) in the ℓ^∞ -norm, in other words, f is $\Theta(1/n)$ -strongly convex around its minimizer. Lemma 17.4.5 summarizes these lemmas: it gives a quantitative bound on the distance to a minimizer, in terms of the gradient.

Lemma 17.4.1. Let $\mathbf{A} \in \mathbb{R}_{\geq 0}^{n \times n}$ with $\|\mathbf{A}\|_1 = 1$ and let $f: V \subset \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ be the potential for this matrix as given in (13.1.3), where V is the orthogonal complement of $(\mathbf{1}_n, -\mathbf{1}_n)$. Then $\text{Hess } f(x, y) \geq \mu(x, y) \cdot n \cdot \mathbf{P}_V$ where \mathbf{P}_V is the projection onto V and $\mu(x, y)$ is the smallest entry appearing in $\mathbf{A}(x, y)$. In particular, f is strictly convex on V .

Proof. The Hessian of the potential $f(x, y) = \sum_{i,j=1}^n A_{ij}e^{x_i+y_j} - \langle r, x \rangle - \langle c, y \rangle$ is given by

$$\text{Hess } f(x, y) = \begin{bmatrix} \text{diag}(r(\mathbf{A}(x, y))) & \mathbf{A}(x, y) \\ \mathbf{A}(x, y)^\top & \text{diag}(c(\mathbf{A}(x, y))) \end{bmatrix}.$$

We give a lower bound on the non-zero eigenvalues of the Hessian as follows. Conjugating the Hessian with the $2n \times 2n$ matrix $\text{diag}(\mathbf{I}, -\mathbf{I})$ preserves the spectrum and yields the matrix

$$\begin{bmatrix} \text{diag}(r(\mathbf{A}(x, y))) & -\mathbf{A}(x, y) \\ -\mathbf{A}(x, y)^\top & \text{diag}(c(\mathbf{A}(x, y))) \end{bmatrix}.$$

One can recognize this as a weighted Laplacian of a complete bipartite graph. We denote by $\mu(\mathbf{x}, \mathbf{y})$ the smallest entry of $\mathbf{A}(\mathbf{x}, \mathbf{y})$ and we use \mathbf{J} for the $n \times n$ all-ones matrix. Then

$$\begin{bmatrix} \text{diag}(\mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) & -\mathbf{A}(\mathbf{x}, \mathbf{y}) \\ -\mathbf{A}(\mathbf{x}, \mathbf{y})^\top & \text{diag}(\mathbf{c}(\mathbf{A}(\mathbf{x}, \mathbf{y}))) \end{bmatrix} \geq \begin{bmatrix} n\mu(\mathbf{x}, \mathbf{y})\mathbf{I} & -\mu(\mathbf{x}, \mathbf{y})\mathbf{J} \\ -\mu(\mathbf{x}, \mathbf{y})\mathbf{J} & n\mu(\mathbf{x}, \mathbf{y})\mathbf{I} \end{bmatrix} = \mu(\mathbf{x}, \mathbf{y}) \begin{bmatrix} n\mathbf{I} & -\mathbf{J} \\ -\mathbf{J} & n\mathbf{I} \end{bmatrix},$$

where the PSD inequality follows because the difference of the terms is the weighted Laplacian of the bipartite graph with weighted bipartite adjacency matrix $\mathbf{A}(\mathbf{x}, \mathbf{y}) - \mu(\mathbf{x}, \mathbf{y})\mathbf{J}$, which has non-negative entries. Now observe that the last term $\begin{bmatrix} n\mathbf{I} & -\mathbf{J} \\ -\mathbf{J} & n\mathbf{I} \end{bmatrix}$ is the (unweighted) Laplacian of the complete bipartite graph $K_{n,n}$, whose spectrum is $2n, n, 0$ with multiplicities $1, 2n - 2$ and 1 respectively. The zero eigenvalue corresponds to the all-ones vector of length $2n$ and it is easy to see that indeed $(\mathbf{1}, -\mathbf{1})$ also lies in the kernel of $\text{Hess } f(\mathbf{x}, \mathbf{y})$. This shows that the non-zero eigenvalues of $\text{Hess } f(\mathbf{x}, \mathbf{y})$ are at least $n \cdot \mu(\mathbf{x}, \mathbf{y})$, and that it has a one-dimensional eigenspace corresponding to 0 , spanned by the vector $(\mathbf{1}, -\mathbf{1})$. Hence, $\text{Hess } f(\mathbf{x}, \mathbf{y}) \geq \mu(\mathbf{x}, \mathbf{y}) \cdot n \cdot \mathbf{P}_V$. \square

We now bound the smallest entry of the rescaled matrix. The main tool for this is Lemma 14.2.1.

Lemma 17.4.2. *Let $\mathbf{A} \in [\mu, \nu]^{n \times n}$ be an entrywise-positive matrix with $\|\mathbf{A}\|_1 = 1$ and let $f: V \subset \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ be the potential for this matrix as given in (13.1.3), where V is the orthogonal complement of $(\mathbf{1}_n, -\mathbf{1}_n)$. Let $(\mathbf{x}^*, \mathbf{y}^*) \in V$ be the unique minimizer of f in V . Then $\mu(\mathbf{x}^*, \mathbf{y}^*) \geq \frac{1}{n^2} \left(\frac{\mu}{\nu}\right)^3$. Moreover, for any $(\mathbf{x}, \mathbf{y}) \in V$ we have $\mu(\mathbf{x}, \mathbf{y}) \geq \mu(\mathbf{x}^*, \mathbf{y}^*)e^{-2\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty}$.*

Proof. By Lemma 17.4.1 f is strictly convex on V . We also know that \mathbf{A} is exactly scalable. Hence f has a unique minimizer $(\mathbf{x}^*, \mathbf{y}^*)$. By Lemma 14.2.1 we know that the variation norms of \mathbf{x}^* and \mathbf{y}^* are bounded by $\ln(\nu/\mu)$. Hence, for every $i, i', j, j' \in [n]$ we have

$$\left| \ln \left(\frac{e^{x_i^* + y_j^*}}{e^{x_{i'}^* + y_{j'}^*}} \right) \right| \leq |x_i^* - x_{i'}^*| + |y_j^* - y_{j'}^*| = 2 \ln(\nu/\mu).$$

Therefore, the ratio between entries of $\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)$ is bounded:

$$\left| \frac{\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)_{ij}}{\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)_{i'j'}} \right| \leq \left| \frac{A_{ij}}{A_{i'j'}} \right| \left| \left(\frac{e^{x_i^* + y_j^*}}{e^{x_{i'}^* + y_{j'}^*}} \right) \right| \leq \frac{\nu}{\mu} e^{2 \ln(\nu/\mu)} = \left(\frac{\nu}{\mu} \right)^3.$$

Since the sum of the entries of $\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)$ equals 1, this implies that the smallest entry of $\mathbf{A}(\mathbf{x}^*, \mathbf{y}^*)$ is at least $\mu(\mathbf{x}^*, \mathbf{y}^*) \geq \frac{1}{n^2} \left(\frac{\mu}{\nu}\right)^3$. Finally, for any $(\mathbf{x}, \mathbf{y}) \in V$ and any $i, j \in [n]$ we have

$$A_{ij} e^{x_i + y_j} \geq A_{ij} e^{x_i^* + y_j^* - 2\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty}$$

which shows $\mu(\mathbf{x}, \mathbf{y}) \geq \mu(\mathbf{x}^*, \mathbf{y}^*)e^{-2\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty}$. \square

Finally, to obtain a diameter bound for the set of points with a small gradient we will use the following (well-known) lemma.

Lemma 17.4.3. Assume $g: \mathbb{R}^d \rightarrow \mathbb{R}$ is a C^2 convex function such that $\text{grad } g(\mathbf{0}) = \mathbf{0}$, and assume that for all $\mathbf{x} \in \mathbb{R}^d$ with $\|\mathbf{x}\|_2 \leq r$, we have $\text{Hess } g(\mathbf{x}) \geq \lambda I$. Then

$$\|\text{grad } g(\mathbf{x})\|_2 \geq \lambda \min(\|\mathbf{x}\|_2, r).$$

Therefore, to guarantee that $\|\mathbf{x}\|_2 \leq C$ for $C \geq 0$, it suffices to show that $\|\text{grad } g(\mathbf{x})\|_2 < \lambda \min(C, r)$ (note that the strict inequality is necessary here because it forces $\min(\|\mathbf{x}\|_2, r) = \|\mathbf{x}\|_2$).

Corollary 17.4.4. If $\text{Hess } g(\mathbf{x}) \geq \lambda I$ holds whenever $\|\mathbf{x}\|_\infty \leq r$, then for all \mathbf{x} , $\|\text{grad } g(\mathbf{x})\|_2 < \lambda \min(C, r)$ implies $\|\mathbf{x}\|_\infty \leq C$.

Proof of Lemma 17.4.3. Fix $\mathbf{x} \in \mathbb{R}^n$ and consider $h: \mathbb{R} \rightarrow \mathbb{R}$ defined by $h(t) = g(t\mathbf{x})$. Then h is convex, $\partial_{t=0}h(t) = 0$ and $\partial_{t=s}^2 h(t) \geq 0$ for all $s \in \mathbb{R}$. Now assume for $s \in \mathbb{R}$ that $|s|\|\mathbf{x}\|_2 \leq r$. Then

$$\begin{aligned} \partial_{t=s}^2 h(t) &= \partial_{t=s}(Dg(t\mathbf{x})[\mathbf{x}]) \\ &= D^2g(s\mathbf{x})[\mathbf{x}, \mathbf{x}] = \mathbf{x}^T \text{Hess } g(s\mathbf{x})\mathbf{x} \geq \lambda\|\mathbf{x}\|_2^2. \end{aligned}$$

We use this to further estimate, for $s \geq 0$, that

$$\begin{aligned} \langle \text{grad } g(s\mathbf{x}), \mathbf{x} \rangle &= \partial_{t=s}h(t) = \int_0^s \partial_{t=\tau}^2 h(t) \, d\tau \\ &\geq \int_0^{\min(s, r/\|\mathbf{x}\|_2)} \partial_{t=\tau}^2 h(t) \, d\tau \\ &\geq \int_0^{\min(s, r/\|\mathbf{x}\|_2)} \lambda\|\mathbf{x}\|_2^2 \, d\tau \\ &= \lambda\|\mathbf{x}\|_2^2 \min(s, r/\|\mathbf{x}\|_2), \end{aligned}$$

where the first inequality follows from the convexity of h . Setting $s = 1$ and using the Cauchy–Schwarz inequality gives

$$\|\text{grad } g(\mathbf{x})\|_2 \|\mathbf{x}\|_2 \geq \lambda\|\mathbf{x}\|_2^2 \min(1, r/\|\mathbf{x}\|_2)$$

so

$$\|\text{grad } g(\mathbf{x})\|_2 \geq \lambda\|\mathbf{x}\|_2 \min(1, r/\|\mathbf{x}\|_2) = \lambda \min(\|\mathbf{x}\|_2, r). \quad \square$$

Lemma 17.4.5. Let $\mathbf{A} \in [\mu, \nu]^{n \times n}$ be an entrywise non-negative matrix with $\|\mathbf{A}\|_1 = 1$ and let $f: V \subset \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ be the potential for this matrix as given in (13.1.3), where V is the orthogonal complement of $(\mathbf{1}_n, -\mathbf{1}_n)$. Let $(\mathbf{x}^*, \mathbf{y}^*)$ be the unique minimizer of f in V and let $0 < \delta < 1$. Let $(\mathbf{x}, \mathbf{y}) \in V$ be such that $\|\nabla f(\mathbf{x}, \mathbf{y})\|_2 < \delta \cdot \frac{1}{n} \left(\frac{\mu}{\nu}\right)^3 e^{-2}$. Then $\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq \delta$.

Proof. Lemma 17.4.1 shows that $\text{Hess } f(\mathbf{x}, \mathbf{y}) \geq n \cdot \mu(\mathbf{x}, \mathbf{y}) \cdot \mathbf{P}_V$, where \mathbf{P}_V is the orthogonal projector on V . Lemma 17.4.2 shows that

$$\mu(\mathbf{x}, \mathbf{y}) \geq \mu(\mathbf{x}^*, \mathbf{y}^*) e^{-2\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty} \geq \frac{1}{n^2} \left(\frac{\mu}{\nu}\right)^3 e^{-2\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty}.$$

Hence, for (\mathbf{x}, \mathbf{y}) with $\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq 1$, we have $\text{Hess } f(\mathbf{x}, \mathbf{y}) \geq \frac{1}{n} \left(\frac{\mu}{\nu}\right)^3 e^{-2} \cdot \mathbf{P}_V$. It then follows from Corollary 17.4.4 that if $\|\nabla f(\mathbf{x}, \mathbf{y})\|_2 < \delta \cdot \frac{1}{n} \left(\frac{\mu}{\nu}\right)^3 e^{-2}$, then $\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq \delta$. \square

17. Quantum query lower bounds: high precision

Observe that for \mathbf{A}^σ the ratio between its largest and smallest entry is $\frac{b+1}{b-1} \leq 3$. This gives the following corollary.

Corollary 17.4.6. *Let \mathbf{A}^σ be as in Section 17.2 and let f be the associated potential. Let $(\mathbf{x}^*, \mathbf{y}^*)$ be the unique exact scaling of \mathbf{A}^σ in V . If $(\mathbf{x}, \mathbf{y}) \in V$ is such that $\|\nabla f(\mathbf{x}, \mathbf{y})\|_2 \leq \frac{\delta}{27ne^2}$, then $\|(\mathbf{x}, \mathbf{y}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq \delta$.*

17.5. Concluding the lower bound for matrix scaling

Let $(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \in V$ be the unique vector such that $(\bar{\mathbf{x}}, \bar{\mathbf{y}}) - (\mathbf{x}, \mathbf{y})$ is a multiple of $(\mathbf{1}_n, -\mathbf{1}_n)$, where (\mathbf{x}, \mathbf{y}) are the scaling vectors of the first step of Sinkhorn. By choosing t and b appropriately we obtain, with high probability over the choice of permutations, a bound on the distance between $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and the unique scaling vectors $(\mathbf{x}^*, \mathbf{y}^*) \in V$ of an exact scaling of \mathbf{A}^σ . This allows us to conclude that, with high probability, all sufficiently precise scalings of \mathbf{A}^σ encode the Hamming weights a_i .

Corollary 17.5.1. *There exists a constant $C > 0$ such that for $b = C\sqrt{\ln n}$ the following holds. With probability $\geq 2/3$ (over the choice of σ) we have for the exact scaling vectors $(\mathbf{x}^*, \mathbf{y}^*) \in V$ of \mathbf{A}^σ that*

$$a_i = \text{sign}(x_{2i}^* - x_{2i-1}^*) \quad \text{for all } i.$$

Furthermore, there exists a constant $C' > 0$ such that for any $(\mathbf{x}', \mathbf{y}')$ that yield a (C'/n^2b) - ℓ^2 -scaling of \mathbf{A}^σ , a_i can be recovered from \mathbf{x}' as $a_i = \text{sign}(x_{2i} - x_{2i-1}) = \text{sign}(x'_{2i} - x'_{2i-1})$.

Proof. Applying Corollary 17.3.5 with $t = 10\sqrt{\ln n}$ shows that with probability at least $2/3$ we have $\|\text{grad } f(\bar{\mathbf{x}}, \bar{\mathbf{y}})\|_2 = \|\text{grad } f(\mathbf{x}, \mathbf{y})\|_2 = \frac{t}{b} \frac{2\sqrt{n}}{b(n^2-4/b^2)\sqrt{k}}$. Hence, there exists a constant $C > 0$ such that for $b = Ct$ we have

$$\|\text{grad } f(\bar{\mathbf{x}}, \bar{\mathbf{y}})\|_2 \leq \frac{1}{nb} \frac{1}{27ne^2}.$$

Corollary 17.4.6 then implies that $\|(\bar{\mathbf{x}}, \bar{\mathbf{y}}) - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq \frac{1}{nb}$ and hence $|(x_{2i-1}^* - x_{2i}^*) - (x_{2i-1} - x_{2i})| \leq \frac{2}{nb}$. Together with Lemma 17.2.1 (which shows that $|x_{2i-1} - x_{2i}| \geq \frac{4}{nb}$) this means that $a_i = \text{sign}(x_{2i}^* - x_{2i-1}^*)$. Moreover, $|x_{2i-1}^* - x_{2i}^*| \geq \frac{2}{nb}$.

Now consider approximate scalings of \mathbf{A}^σ . Without loss of generality we may assume that the $(\mathbf{x}', \mathbf{y}')$ that yield a $(\frac{1}{2nb} \frac{1}{27ne^2})$ - ℓ^2 -scaling of \mathbf{A}^σ belong to V (otherwise we shift it by an appropriate multiple of $(\mathbf{1}_n, -\mathbf{1}_n)$). Then, again due to Corollary 17.4.6, we obtain that $\|(\mathbf{x}', \mathbf{y}') - (\mathbf{x}^*, \mathbf{y}^*)\|_\infty \leq \frac{1}{2nb} \leq \frac{1}{4}|x_{2i-1}^* - x_{2i}^*|$ and hence $|(x'_{2i-1} - x'_{2i}) - (x_{2i-1}^* - x_{2i}^*)| \leq \frac{1}{2}|x_{2i-1}^* - x_{2i}^*|$ which means that $\text{sign}(x'_{2i} - x'_{2i-1}) = \text{sign}(x_{2i}^* - x_{2i-1}^*) = a_i$. \square

Theorem 17.5.2. *There exists a constant $C > 0$ such that every matrix scaling algorithm that, with probability $\geq \frac{3}{2} \exp(-n/100)$, finds scalings for $n \times n$ -matrices with ℓ^2 -error $C/(n^2\sqrt{\ln n})$ must make at least $\Omega(n^2)$ queries to the matrix. This even holds for uniform targets and entrywise-positive matrices with smallest entry $\Omega(1/n^2)$.*

Proof. We construct a set of hard instances as in Section 17.2. Let $n \geq 4$ be even. Let $k = n/2$ and let $z^1, \dots, z^k \in \{\pm 1\}^n$ have Hamming weight $|z^i| = |\{j : z_j^i = 1\}| =$

$n/2 + \alpha_i$ for $\alpha_i \in \{\pm 1\}$. By Theorem 17.1.1, finding at least 99% of the α_i 's with probability $\geq \exp(-n/100)$ takes $\Omega(n^2)$ -queries to the z_j^i . One can recover the α_i 's with probability $\geq 2/3$ as follows. First, sample the $\sigma^1, \dots, \sigma^{n/2}$ uniformly from S_n . A single query to \mathbf{A}^σ takes a single query to some w^i , which takes a single query to z^i . Using Corollary 17.5.1, there exists a constant $C > 0$ such that, with probability $\geq 2/3$, any scaling of \mathbf{A}^σ with ℓ^2 -error $C/(n^2\sqrt{\ln n})$ recovers all α_i 's. Therefore any matrix scaling algorithm finding such a scaling with probability $\geq \exp(-n/100)$ allows us to find all α_i 's with probability $\geq \exp(-n/100)$. \square

Corollary 17.5.3. *There exist constants $C_0, C_1 > 0$ such that any matrix scaling algorithm that, with probability $\geq \exp(-C_0 n/\ln(n))$, finds scalings for $n \times n$ -matrices with at most m non-zero entries and ℓ^2 -error $C_1/(m\sqrt{\ln(m/n)})$ must make at least $\tilde{\Omega}(m)$ queries to the matrix. This even holds for uniform targets and matrices with smallest non-zero entry $\Omega(1/m)$.*

Proof. We construct a set of sparse hard instances by taking direct sums of the hard instances used in the proof of Theorem 17.5.2. Concretely, let $s \geq 4$ be even and assume that n is a multiple of s . Let $\mathbf{A}_1^{\sigma_1}, \dots, \mathbf{A}_{n/s}^{\sigma_{n/s}} \in [0, 1]^{s \times s}$ be n/s independently drawn hard instances from the set constructed in the proof of Theorem 17.5.2. We use this to create a sparse instance $\mathbf{A} = \frac{s}{n} \oplus_{i=1}^{n/s} \mathbf{A}_i^{\sigma_i}$. Note that $\|\mathbf{A}\|_1 = 1$ and each row of \mathbf{A} has exactly s non-zero entries (which means $m = ns$). Let (\mathbf{x}, \mathbf{y}) be an ε - ℓ^2 -scaling of \mathbf{A} to uniform marginals. Then we have

$$\varepsilon^2 \geq \left\| \frac{\mathbf{1}}{n} - \mathbf{r}(\mathbf{A}(\mathbf{x}, \mathbf{y})) \right\|_2^2 = \sum_{i=1}^{n/s} \left\| \frac{\mathbf{1}}{n} - \frac{s}{n} \mathbf{A}_i^{\sigma_i}(\mathbf{x}|_i, \mathbf{y}|_i) \right\|_2^2 = \sum_{i=1}^{n/s} \left(\frac{s}{n} \right)^2 \left\| \frac{\mathbf{1}}{s} - \mathbf{A}_i^{\sigma_i}(\mathbf{x}|_i, \mathbf{y}|_i) \right\|_2^2$$

where $(\mathbf{x}|_i, \mathbf{y}|_i)$ is the restriction of (\mathbf{x}, \mathbf{y}) to the coordinates corresponding to the i th block. In particular, for each $i \in [n/s]$, the pair $(\mathbf{x}|_i, \mathbf{y}|_i)$ forms an $\frac{\varepsilon n}{s}$ - ℓ^2 -scaling of $\mathbf{A}_i^{\sigma_i}$ to marginals $\mathbf{1}/s$. Hence, for $\varepsilon = C/(ns\sqrt{\ln(s)})$ we recover for each block a scaling with ℓ^2 -error $C/(s^2\sqrt{\ln(s)})$. For each block this allows us, with probability $\geq 2/3$ over the choice of σ_i , to compute the Hamming weights of the associated bit strings. Hence, for a suitably large constant c_0 , using $c_0 \ln(n)$ successful runs of the scaling algorithm with independently drawn choices of the σ_i 's allows us to compute the Hamming weights of all n bit strings with probability at least $2/3$. The probability that all the runs of the scaling algorithm are successful is at least $(\exp(-C_0 n/\ln(n)))^{c_0 \ln(n)} = \exp(-C_0 c_0 n) \geq \frac{3}{2} \exp(-n/100)$, where the last inequality determines the choice of C_0 . Hence, we compute the Hamming weights of all n bit strings with probability at least $\exp(-n/100)$ and by Theorem 17.1.1 this requires at least $\Omega(ns)$ quantum queries to the bit strings. \square

17.6. Lower bound for matrix balancing

We now show how to obtain an $\Omega(m)$ quantum query lower bound for ε - ℓ^2 -matrix balancing for $\varepsilon = C/m$ for a suitably small constant $C > 0$. We first show how to do so in the dense case where the support of the matrix equals the complete bipartite graph $K_{n,n}$. Our lower bound is again based on a reduction to the problem of counting Hamming weights of bit strings, i.e., it is based on the lower bound

shown in Theorem 17.1.1. More concretely, let $n \geq 2$ be even, set $k = n/2$, and let $z^1, \dots, z^k \in \{\pm 1\}^n$ be bit strings of length n with the Hamming weight of z^j equal to $\frac{n}{2} + a_j$ for $a_j \in \{\pm 1\}$. We construct a single matrix balancing instance of size $2n \times 2n$ that has the property that all balancing-factors \mathbf{x} that provide a sufficiently balanced matrix encode the bits a_1, \dots, a_k . Let \mathbf{A} be the n -by- n matrix that, for $j \in [k]$, contains the vector $1 + z^j/2$ on the $2j - 1$ th row and $1 - z^j/2$ on the $2j$ th row. All column-marginals of \mathbf{A} are equal to n . Moreover $r_{2j-1}(\mathbf{A}) = n + a_j$ and $r_{2j}(\mathbf{A}) = n - a_j$ for each $j \in [k]$. We now consider the matrix balancing instance given by \mathbf{B} where

$$\mathbf{B} := \begin{bmatrix} 0 & \mathbf{A} \\ 2 \cdot \mathbf{1}_{n \times n} - \mathbf{A}^\top & 0 \end{bmatrix}. \quad (17.6.1)$$

Note that the bottom-left block corresponds to the transpose of the matrix that is obtained in a similar manner as \mathbf{A} starting from the *negated bit strings* $-z^1, \dots, -z^k$. We thus have the following equalities for each $j \in [k]$:

$$\begin{aligned} r_{2j-1}(\mathbf{B}) &= c_{2j}(\mathbf{B}) = r_{2j-1}(\mathbf{A}) = n + a_j, \\ r_{2j}(\mathbf{B}) &= c_{2j-1}(\mathbf{B}) = r_{2j}(\mathbf{A}) = n - a_j, \\ r_{n+2j-1}(\mathbf{B}) &= r_{n+2j}(\mathbf{B}) = c_{n+2j-1}(\mathbf{B}) = c_{n+2j}(\mathbf{B}) = n. \end{aligned}$$

We first show that \mathbf{B} can be exactly balanced by factors X_1, \dots, X_{2n} that moreover encode the bits a_1, \dots, a_k . For $i \in [n]$ define $X_i = \sqrt{c_i(\mathbf{B})/r_i(\mathbf{B})}$. Then for $j \in [k]$ we have

$$X_{2j-1} = X_{2j}^{-1} = \sqrt{\frac{n - a_j}{n + a_j}}. \quad (17.6.2)$$

Clearly X_{2j} and X_{2j-1} encode a_j . Then for $\mathbf{X}^* = \text{diag}(X_1, \dots, X_n, 1, \dots, 1)$ (thus $X_{n+i} = 1$) and $\mathbf{C} = \mathbf{X}^* \mathbf{B} \mathbf{X}^{*-1}$, one has $r_i(\mathbf{C}) = c_i(\mathbf{C})$ for every $i \in [n]$ by construction. We show that in fact \mathbf{C} is exactly balanced.

Proposition 17.6.1. *The matrix $\mathbf{C} = \mathbf{X}^* \mathbf{B} \mathbf{X}^{*-1}$ is exactly balanced.*

Proof. Observe first that

$$\mathbf{C} = \begin{bmatrix} 0 & \mathbf{X}\mathbf{A} \\ (2 \cdot \mathbf{1}_{n \times n} - \mathbf{A}^\top)\mathbf{X}^{-1} & 0 \end{bmatrix}$$

where we use the notation $\mathbf{X} = \text{diag}(X_1, \dots, X_n)$. Note that X_1, \dots, X_n are defined in such a way that the first n rows and columns are exactly balanced (because of the bipartite structure of \mathbf{B} , balancing the i th marginals does not affect the i' th marginals for $i, i' \in [n]$). It remains to verify that $r_{n+i}(\mathbf{C}) = c_{n+i}(\mathbf{C})$ for each $i \in [n]$. Note that for $i \in [n]$, we have $c_{n+i}(\mathbf{C}) = c_i(\mathbf{X}\mathbf{A})$ and

$$\begin{aligned} r_{n+i}(\mathbf{C}) &= r_i((2 \cdot \mathbf{1}_{n \times n} - \mathbf{A}^\top)\mathbf{X}^{-1}) \\ &= c_i(\mathbf{X}^{-1}(2 \cdot \mathbf{1}_{n \times n} - \mathbf{A})) \end{aligned}$$

where the last equality is obtained by transposing. We expand

$$c_i(\mathbf{X}\mathbf{A}) = \sum_{j=1}^k \left(1 + \frac{z_i^j}{2}\right) X_{2j-1} + \left(1 - \frac{z_i^j}{2}\right) X_{2j},$$

$$c_i(\mathbf{X}^{-1}(2 \cdot \mathbf{1}_{n \times n} - \mathbf{A})) = \sum_{j=1}^k \left(1 - \frac{z_i^j}{2}\right) X_{2j-1}^{-1} + \left(1 - \frac{z_i^j}{2}\right) X_{2j}^{-1}.$$

Using the relation $X_{2j} = X_{2j-1}^{-1}$ from Eq. (17.6.2), we thus obtain

$$r_{n+i}(\mathbf{C}) = c_i(\mathbf{X}^{-1}(2 \cdot \mathbf{1}_{n \times n} - \mathbf{A})) = c_i(\mathbf{X}\mathbf{A}) = c_{n+i}(\mathbf{C}).$$

This shows that \mathbf{C} is exactly balanced. \square

We now show that any vector $\mathbf{x} \in V := \{\mathbf{x} \in \mathbb{R}^{2n} : \langle \mathbf{x}, \mathbf{1}_n \rangle = 0\}$ for which $\mathbf{B}(\mathbf{x})$ is ε - ℓ^2 -balanced is close to $\mathbf{x}^* := \ln(\mathbf{X}^*)$, for ε small enough. To do so, we consider the two convex functions

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i,j=1}^{2n} B_{ij} e^{x_i - x_j}, \\ F(\mathbf{x}) &= \log(f(\mathbf{x})). \end{aligned} \tag{17.6.3}$$

Their gradients are $\nabla f(\mathbf{x}) = \mathbf{r}(\mathbf{B}(\mathbf{x})) - \mathbf{c}(\mathbf{B}(\mathbf{x}))$ and $\nabla F(\mathbf{x}) = \frac{\nabla f(\mathbf{x})}{f(\mathbf{x})}$, and therefore \mathbf{x}^* is a minimizer of both f and F . Moreover, if $\mathbf{B}(\mathbf{x})$ is ε - ℓ^2 -balanced, then we have

$$\|\nabla F(\mathbf{x})\|_2 = \frac{\|\mathbf{r}(\mathbf{B}(\mathbf{x})) - \mathbf{c}(\mathbf{B}(\mathbf{x}))\|_2}{\|\mathbf{B}(\mathbf{x})\|_1} \leq \varepsilon.$$

We show that for the matrix balancing instances defined in Eq. (17.6.1), F is moreover strongly convex around its minimizer \mathbf{x}^* .

Proposition 17.6.2. *Let $F(\mathbf{x}) = \ln(\sum_{i,j} B_{ij} e^{x_i - x_j})$ for a matrix \mathbf{B} as in Eq. (17.6.1). Let $\mathbf{x}^* = \ln(\mathbf{X}^*)$ be its minimizer. Then, for all \mathbf{x} with $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq 1/100$, we have*

$$\text{Hess } F(\mathbf{x}) \geq c \cdot n \Pi_V$$

where Π_V is the projector on the complement of the all-ones vector and $0 < c < 1$ is a constant. Moreover, for such \mathbf{x} one has $\|\text{grad } f(\mathbf{x})\|_2 \leq \frac{n}{25}$ and $\frac{n}{2} \Pi_V \leq \text{Hess } f(\mathbf{x}) \leq 4n \Pi_V$.

Proof. We have

$$\begin{aligned} \text{Hess } F(\mathbf{x}) &= \frac{\text{Hess } f(\mathbf{x})}{f(\mathbf{x})} - \frac{1}{f(\mathbf{x})^2} \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^\top \\ &= \frac{\text{diag}(\mathbf{r}(\mathbf{B}(\mathbf{x})) + \mathbf{c}(\mathbf{B}(\mathbf{x}))) - \mathbf{B}(\mathbf{x}) - \mathbf{B}(\mathbf{x})^\top}{\|\mathbf{B}(\mathbf{x})\|_1} - \frac{\mathbf{d}(\mathbf{B}(\mathbf{x})) \mathbf{d}(\mathbf{B}(\mathbf{x}))^\top}{\|\mathbf{B}(\mathbf{x})\|_1^2} \end{aligned}$$

where we define $\mathbf{d}(\mathbf{B}(\mathbf{x})) := \mathbf{r}(\mathbf{B}(\mathbf{x})) - \mathbf{c}(\mathbf{B}(\mathbf{x}))$ and we use

$$\text{Hess } f(\mathbf{x}) = \text{diag}(\mathbf{r}(\mathbf{B}(\mathbf{x})) + \mathbf{c}(\mathbf{B}(\mathbf{x}))) - \mathbf{B}(\mathbf{x}) - \mathbf{B}(\mathbf{x})^\top.$$

To obtain a lower bound on the non-zero eigenvalues of $\text{Hess } F(\mathbf{x})$ we bound the three quantities $\|\mathbf{B}(\mathbf{x})\|_1$, $\text{Hess } f(\mathbf{x})$, and $\|\nabla f(\mathbf{x})\|_2$ separately.

17. Quantum query lower bounds: high precision

First note that for all $\mathbf{x} \in \mathbb{R}^{2n}$ with $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq 1/100$ we have $e^{-2/100} \leq \frac{\mathbf{B}(\mathbf{x})_{i,j}}{\mathbf{B}(\mathbf{x}^*)_{i,j}} \leq e^{2/100}$ for all $i, j \in [2n]$. In particular $\|\mathbf{B}(\mathbf{x})\|_1$ lies within a constant factor of $\|\mathbf{B}(\mathbf{x}^*)\|_1$. We moreover have $\|\mathbf{B}(\mathbf{x}^*)\|_1 = \Theta(n^2)$.

We now consider the Hessian of f . It can be viewed as the Laplacian of a graph on $2n$ vertices with edge weights defined by $\mathbf{B}(\mathbf{x}) + \mathbf{B}(\mathbf{x})^\top$. Given the structure of the matrices \mathbf{B} , the edges with strictly positive weight correspond to the complete bipartite graph $K_{n,n}$. We bound the Hessian of f using the smallest and largest edge weight. More precisely, let w_{\min} and w_{\max} be, respectively, lower and upper bounds on edge weights of $\text{Hess } f(\mathbf{x})$, then

$$w_{\min} \mathcal{L}(K_{n,n}) \leq \text{Hess } f(\mathbf{x}) \leq w_{\max} \mathcal{L}(K_{n,n}),$$

where $\mathcal{L}(K_{n,n})$ is the Laplacian of the *unweighted* complete bipartite graph $K_{n,n}$ (which has spectrum $2n, n, 0$ with multiplicities $1, 2n - 2, 1$). In particular, at \mathbf{x}^* , the non-zero entries of $\mathbf{B}(\mathbf{x}^*)$ are at least $(1 - \frac{1}{2})\sqrt{\frac{n-1}{n+1}}$ and at most $(1 + \frac{1}{2})\sqrt{\frac{n+1}{n-1}}$.

This makes the edge weights at least $w_{\min} = \sqrt{\frac{n-1}{n+1}}$ and at most $w_{\max} = 3\sqrt{\frac{n+1}{n-1}}$. Similarly, for all \mathbf{x} that satisfy $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq 1/100$ the edge weights are all $\Theta(1)$ (since the entries of $\mathbf{B}(\mathbf{x})$ lie within a constant factor of those of $\mathbf{B}(\mathbf{x}^*)$). We thus have the following bounds on $\text{Hess } f(\mathbf{x})$: for all \mathbf{x} that satisfy $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq 1/100$, we have

$$\frac{n}{2} \Pi_V \leq \text{Hess } f(\mathbf{x}) \leq 4n \Pi_V.$$

We finally use the above upper bound on the Hessian of f to upper bound the norm of $\nabla f(\mathbf{x})$. To do so, we require not only a bound on the ℓ^∞ -norm of $\mathbf{x} - \mathbf{x}^*$ but also on its ℓ^2 -norm. More precisely, the above bound implies that $\|\text{grad } f(\mathbf{x})\|_2 \leq \frac{n}{25}$ whenever $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq 1/100$. To see this, observe that

$$\begin{aligned} \|\text{grad } f(\mathbf{x})\|_2^2 &= \langle \text{grad } f(\mathbf{x}), \text{grad } f(\mathbf{x}) \rangle \\ &= \int_0^1 \langle \text{grad } f(\mathbf{x}), \text{Hess } f(t\mathbf{x} + (1-t)\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*) \rangle dt \\ &\leq \|\text{grad } f(\mathbf{x})\|_2 \|\mathbf{x} - \mathbf{x}^*\|_2 \int_0^1 \|\text{Hess } f(t\mathbf{x} + (1-t)\mathbf{x}^*)\|_{\text{op}} dt \\ &\leq \|\text{grad } f(\mathbf{x})\|_2 \|\mathbf{x} - \mathbf{x}^*\|_2 \cdot 4n \end{aligned}$$

so $\|\text{grad } f(\mathbf{x})\|_2 \leq \frac{4n}{100}$ whenever $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \frac{1}{100}$.

Combining the above three bounds we obtain that the smallest non-zero eigenvalue of $\text{Hess } F(\mathbf{x})$ is at least

$$\frac{n/2}{cn^2} - \frac{n^2}{(cn^2)^2} = \Omega(1/n)$$

whenever $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \frac{1}{100}$. Moreover, for such \mathbf{x} the Hessian $\text{Hess } F(\mathbf{x})$ has only a single eigenvalue equal to zero (corresponding to the all-ones eigenvector). \square

The strong convexity of F around the minimizer implies a lower bound on $\|\nabla F(\mathbf{x})\|_2$ for $\mathbf{x} \in V$ that are far from \mathbf{x}^* , via Lemma 17.4.3. In other words, we obtain an upper bound on the distance $\|\mathbf{x} - \mathbf{x}^*\|_2$ for all $\mathbf{x} \in V$ for which $\mathbf{B}(\mathbf{x})$ is approximately balanced.

Proposition 17.6.3. *Let $n \geq 2$ be even, $k = n/2$, and $z^1, \dots, z^k \in \{\pm 1\}^n$ with Hamming weight $n/2 + \alpha_j$ for $\alpha_j \in \{\pm 1\}$. Let $\mathbf{B} \in \mathbb{R}^{2n \times 2n}$ be the corresponding matrix defined in Eq. (17.6.1). Then for any $\mathbf{x} \in V$ such that $e^{\mathbf{x}} \mathbf{B} e^{-\mathbf{x}}$ is $\frac{C}{n^2}$ - ℓ^2 -balanced, we have $\text{sign}(x_{2j-1}) = \alpha_j$ for all $j \in [k]$. Here $C > 0$ is a constant.*

Proof. The strong convexity of F around $\mathbf{x}^* = \ln(\mathbf{X}^*)$ shown in Proposition 17.6.2 gives, by Lemma 17.4.3, the following lower bound on the norm of the gradient

$$\|\nabla F(\mathbf{x})\|_2 \geq \frac{c}{n} \min\{\|\mathbf{x} - \mathbf{x}^*\|_2, 1/100\}$$

for a suitable constant $c > 0$ and $\mathbf{x} \in V$. In particular, for vectors $\mathbf{x} \in V$ that provide an ε - ℓ^2 -balancing, we obtain

$$\varepsilon \geq \|\nabla F(\mathbf{x})\|_2 \geq \frac{c}{n} \min\{\|\mathbf{x} - \mathbf{x}^*\|_2, 1/100\}.$$

For $\varepsilon < c/(100n)$ this shows that $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \varepsilon n/c$ and thus also $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq \varepsilon n/c$. Recall that for $i \in [n]$ we have $|\ln(X_i)| = \ln(\sqrt{\frac{n+1}{n-1}})$ and thus $|\ln(X_i)| > \frac{c'}{n}$ for a suitable constant c' and n large enough. Hence, for $\varepsilon = \frac{c'c}{2n^2}$ we have $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq \frac{c'c}{2n^2} \frac{n}{c} = \frac{c'}{2n}$ and thus for $j \in [k]$ we have $\text{sign}(x_{2j}) = \text{sign}(\ln(X_{2j}^*)) = \alpha_j$. \square

Theorem 17.6.4. *There exists a constant $C > 0$ such that any matrix balancing algorithm that, with probability $\geq \exp(-n/100)$, finds balancings for $n \times n$ -matrices that have $\Omega(n^2)$ non-zero entries with ℓ^2 -error $\frac{C}{n^2}$ must make at least $\Omega(n^2)$ queries to the matrix. This even holds for matrices for which the ratio between largest and smallest entry is a constant, and whose support equals the complete bipartite graph $K_{n/2, n/2}$.*

Proof. We use the set of balancing instances defined in Eq. (17.6.1). By Theorem 17.1.1, finding at least 99% of the α_i 's with probability $\geq \exp(-n/100)$ takes $\Omega(n^2)$ queries to the z_j^i and thus $\Omega(n^2)$ queries to \mathbf{B} . Proposition 17.6.3 shows that there exists a constant $C > 0$ such that every $\mathbf{x} \in V$ for which $\mathbf{B}(\mathbf{x})$ is $\frac{C}{n^2}$ - ℓ^2 -balanced, allows one to recover *all* α_i correctly. Finding such a vector $\mathbf{x} \in V$ thus requires $\Omega(n^2)$ queries to \mathbf{B} . \square

We finally show how to extend the above result to sparse instances \mathbf{B} that have s non-zero entries per row and column. Letting $m = ns$ be the total number of non-zero entries in such instances, we show an $\Omega(m)$ -query lower bound for high enough precision.

Theorem 17.6.5. *There exists a constant $C > 0$ such that any matrix balancing algorithm that, with probability $\geq \exp(-n/100)$, finds balancings for $n \times n$ -matrices that have m non-zero entries with ℓ^2 -error $\frac{C}{m}$ must make at least $\Omega(m)$ queries to the matrix. This even holds for matrices for which the ratio between largest and smallest entry is a constant, and which have exactly $s \geq 4$ non-zero entries per row and column.*

Proof. We consider $k = n/s$ many $s \times s$ matrices $\mathbf{A}^1, \dots, \mathbf{A}^k$ where each \mathbf{A}^i is associated to an $s/2$ -tuple of s -bit strings as before. Construct \mathbf{B}^i from \mathbf{A}^i as before and define $\mathbf{B} = \bigoplus_{i \in [k]} \mathbf{B}^i$. Then $\mathbf{B} \in \mathbb{R}^{2n \times 2n}$. We first show $F_{\mathbf{B}}(\mathbf{x}) = \log(\|\mathbf{B}(\mathbf{x})\|_1)$ is strongly convex around its minimizer \mathbf{x}^* when restricted to the linear subspace $\bigoplus_{i \in [k]} V_s$, where $V_s := \{\mathbf{y} \in \mathbb{R}^{2s} \mid \langle \mathbf{y}, \mathbf{1}_{2s} \rangle = 0\}$. Let us use $x|_i$ to denote the

17. Quantum query lower bounds: high precision

vector \mathbf{x} restricted to the coordinates corresponding to the i th block. We will use the two potentials defined in Eq. (17.6.3) for each of the blocks, we denote them using $f_{\mathbf{B}^i}$ and $F_{\mathbf{B}^i}$. We have the following two identities:

$$\begin{aligned}\nabla f_{\mathbf{B}}(\mathbf{x}) &= \oplus_{i \in [k]} \nabla f_{\mathbf{B}^i}(\mathbf{x}|_i), \\ \text{Hess } f_{\mathbf{B}}(\mathbf{x}) &= \oplus_{i \in [k]} \text{Hess } f_{\mathbf{B}^i}(\mathbf{x}|_i).\end{aligned}$$

Let $\mathbf{x}^* := \oplus_{i \in [k]} \mathbf{x}^*|_i$ with $\mathbf{x}^*|_i$ the minimizer of $F_{\mathbf{B}^i}$. When $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq 1/100$ we have $\|\mathbf{x}|_i - \mathbf{x}^*|_i\|_2 \leq 1/100$ for each $i \in [k]$. Hence, by Proposition 17.6.2, for such \mathbf{x} we have $\|\text{grad } f_{\mathbf{B}^i}(\mathbf{x}|_i)\|_2 \leq \frac{s}{25}$ and $\frac{s}{2}\Pi_{V_s} \leq \text{Hess } f_{\mathbf{B}^i}(\mathbf{x}|_i) \leq 4s\Pi_{V_s}$. In particular, for such \mathbf{x} we have

$$\|\nabla f_{\mathbf{B}}(\mathbf{x})\|_2^2 \leq ks^2/25^2.$$

For all \mathbf{x} with $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq 1/100$, one has $\|\mathbf{B}(\mathbf{x})\|_1 \in [e^{-2/100}, e^{2/100}]\|\mathbf{B}(\mathbf{x}^*)\|_1$ and $\|\mathbf{B}(\mathbf{x}^*)\|_1 = \Theta(ks^2) = \Theta(m)$. Hence, for $F_{\mathbf{B}}$ we have

$$\begin{aligned}\text{Hess } F_{\mathbf{B}}(\mathbf{x}) &= \frac{\oplus_{i \in [k]} \text{Hess } f_{\mathbf{B}^i}(\mathbf{x}|_i)}{\|\mathbf{B}(\mathbf{x})\|_1} - \frac{\nabla f_{\mathbf{B}}(\mathbf{x})\nabla f_{\mathbf{B}}(\mathbf{x})^\top}{\|\mathbf{B}(\mathbf{x})\|_1^2} \\ &\geq \frac{\oplus_{i \in [k]} \frac{s}{2}\Pi_{V_s}}{\|\mathbf{B}(\mathbf{x})\|_1} - \frac{\frac{ks^2}{25^2}I_n}{\|\mathbf{B}(\mathbf{x})\|_1^2}\end{aligned}$$

which implies that the smallest non-zero eigenvalue of $\text{Hess } F_{\mathbf{B}}(\mathbf{x})$ is at least

$$\frac{s/2}{\|\mathbf{B}(\mathbf{x})\|_1} - \frac{ks^2}{25\|\mathbf{B}(\mathbf{x})\|_1^2} = \frac{\|\mathbf{B}(\mathbf{x})\|_1 s/2 - ks^2/25^2}{\|\mathbf{B}(\mathbf{x})\|_1^2} = \Omega\left(\frac{1}{ks}\right) = \Omega(1/n)$$

where we use that $\|\mathbf{B}(\mathbf{x})\|_1 \in [\frac{1}{2}ks^2, 4ks^2]$ and $n = ks$. Indeed, to compute the 1-norm of $\mathbf{B}(\mathbf{x})$, observe that all its $2ks^2$ non-zero entries take values in $[1/4, 2]$ since

$$\min_{i,j} B_{ij} e^{x_i - x_j} \geq e^{-2/100} \min_{i,j} B_{ij} e^{x_i^* - x_j^*} = e^{-2/100} \frac{1}{2} \sqrt{\frac{s-1}{s+1}} \geq 1/4$$

and similarly

$$\max_{i,j} B_{ij} e^{x_i - x_j} \leq e^{2/100} \max_{i,j} B_{ij} e^{x_i^* - x_j^*} = e^{2/100} \left(1 + \frac{1}{2}\right) \sqrt{\frac{s+1}{s-1}} \leq 2.$$

We then proceed as in the proof of Proposition 17.6.3. We may assume $\mathbf{x} \in \oplus_{i \in [k]} V_s$. Then, by Lemma 17.4.3, for vectors $\mathbf{x} \in \oplus_{i \in [k]} V_s$ that provide an ε - ℓ^2 -balancing, we obtain

$$\varepsilon \geq \|\nabla F(\mathbf{x})\|_2 \geq \frac{c}{n} \min\{\|\mathbf{x} - \mathbf{x}^*\|_2, 1/100\}.$$

For $\varepsilon < c/(100n)$ this shows that $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \varepsilon n/c$ and thus also $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq \varepsilon n/c$. Recall that for $i \in [n]$ we have $|\ln(X_i)| = \ln\left(\sqrt{\frac{s+1}{s-1}}\right)$ and thus $|\ln(X_i)| > \frac{c'}{s}$ for a suitable constant c' and s large enough. Hence, for $\varepsilon = \frac{c'c}{2ns}$ we have $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq \frac{c'c}{2ns} \frac{n}{c} = \frac{c'}{2s}$ and thus for $j \in [ks/2]$ we have $\text{sign}(x_{2j}) = \text{sign}(\ln(X_{2j}^*)) = \alpha_j$. An ε - ℓ^2 -balancing thus allows us to recover the Hamming weight of the $ks/2$ many s -bit strings. By Theorem 17.1.1 this requires $\Omega(ks^2) = \Omega(m)$ queries to \mathbf{B} . \square

Remark 17.6.6. A natural question is whether one can improve the $\text{poly}(1/\varepsilon)$ -dependence of our sublinear matrix scaling and matrix balancing algorithms. Theorems 17.5.2 and 17.6.5 and Corollary 17.5.3 show that a $\tilde{O}(\sqrt{mn}/\varepsilon^c)$ time quantum algorithm for either the ε - ℓ^2 -scaling problem or the ε - ℓ^2 -balancing problem requires $c \geq 1/4 - o(1)$. For example, Corollary 17.5.3 shows that an $\frac{C}{m \log(m/n)}$ - ℓ^2 -scaling requires $\tilde{\Omega}(m)$ queries and thus $(m \log(m/n))^c \geq \frac{m}{\sqrt{mn} \text{polylog}(n)}$. Taking $m = \Theta(n^2)$, this implies $c \geq 1/4 - o(1)$. For ε - ℓ^2 -balancing one similarly, via Theorem 17.6.5, obtains $c \geq 1/4 - o(1)$.

17.7. Lower bound for computing the row marginals

In this section we show that computing an ε - ℓ^1 -approximation of the row (or column marginals) of an entrywise-positive $n \times n$ matrix with 1-norm at most 1 takes $\Omega(n^{1.5}/\sqrt{\varepsilon})$ queries to its entries (for $\varepsilon = \Omega(1/n)$). This complements the upper bounds of Corollaries 12.4.4 and 13.3.1. as a consequence, the same lower bound holds for computing an approximation of the gradient of the common (convex) potential functions used for matrix scaling and balancing – among which are the potentials we use in Chapters 14 and 15 – takes as many queries. Although the bound does not imply that testing whether a matrix is ε - ℓ^1 -scaled takes at least $\Omega(n^{1.5}/\sqrt{\varepsilon})$ queries, it gives reasonable evidence that this should be the case.

Theorem 17.7.1. Let $s, n \geq 1$ be integers, let $\mathbf{A} \in \{0, 1\}^{n \times s}$, and let $1 < k < s$. Then there is a $\delta = \Omega(1/k)$ such that $\Omega\left(n\sqrt{s/\delta}\right)$ quantum queries to a dense oracle for \mathbf{A} are required in general to (with probability at least $\exp(-n/100)$) find a $\tilde{\mathbf{r}}$ such that $\|\tilde{\mathbf{r}} - \mathbf{r}(\mathbf{A})\|_1 \leq \delta \|\mathbf{A}\|_1$.

Proof. Without loss of generality we shall assume that n is even. Let $x \in \{0, 1\}^n$ be a bit string with $|x| = n/2$, and let $z^1, \dots, z^n \in \{0, 1\}^s$ be bit strings with $|z^i| = k + (-1)^{x_i}$. Let $\mathbf{A} \in \{0, 1\}^{n \times s}$ be the matrix whose i -th row is given by z^i . Then $\|\mathbf{A}\|_1 = nk$ by construction.

As $r_i(\mathbf{A}) = |z^i|$, learning r_i suffices to learn x_i . Doing so for a single i requires $\Omega(\sqrt{sk})$ quantum queries by [NW99, Cor. 1.2]. We now apply the strong direct product theorem of Lee and Roland [LR13, Thm. 5.4]. Let $\mathcal{D} = \{z \in \{0, 1\}^s : |z| = k - 1 \text{ or } k + 1\}$, and define the partial Boolean function $f: \mathcal{D} \rightarrow \{\pm 1\}$ by $f(z) = |z| - k$. Let $f^{(n)}: \mathcal{D}^n \rightarrow \{\pm 1\}^k$ be defined by $f^{(n)}(z^1, \dots, z^n) = (f(z^1), \dots, f(z^n))$. Then by the strong direct product theorem, every quantum algorithm that with probability $\geq \exp(-n/100)$ computes a bit string $\tilde{a} \in \{\pm 1\}^n$ which agrees with $f^{(n)}(z^1, \dots, z^n)$ in at least 99% of their entries, must use at least $\Omega(n\sqrt{sk})$ queries (passing through the general adversary quantity as in the proof of Theorem 17.1.1). In other words, for $\lambda = 0.01$, learning a $(1 - \lambda)$ -fraction of the x_i 's with probability $\geq \exp(-n/100)$ requires $\Omega(n\sqrt{sk})$ quantum queries to \mathbf{A} .

It remains to show that there is a $\delta = \Omega(1/k)$ such that solving computing a $\delta \|\mathbf{A}\|_1$ - ℓ^1 -approximation $\tilde{\mathbf{r}}$ of $\mathbf{r}(\mathbf{A})$ suffices to solve this problem. Observe that such an approximation satisfies that at most δnk indices $i \in [n]$ satisfy $|\tilde{r}_i - r_i(\mathbf{A})| \geq 1$, by virtue of $\|\mathbf{A}\|_1 = nk$. Therefore for $n - \delta nk$ rows, we would learn x_i . In other words, whenever $\delta k \leq \lambda$, computing such a $\tilde{\mathbf{r}}$ requires $\Omega(n\sqrt{sk})$ quantum queries to \mathbf{A} . \square

17. Quantum query lower bounds: high precision

The above argument can also be adapted to yield instances \mathbf{A} which are entrywise positive. To achieve this, consider instead the matrix whose (i, j) -th entry is $\frac{1}{2}(\frac{k}{s} + z_j^i)$. Then the row sums r_i are $\frac{1}{2} \cdot \frac{k}{s} \cdot s + \frac{1}{2}(k + (-1)^{x_i}) = k + \frac{(-1)^{x_i}}{2}$ instead, and the matrix still has sum of entries equal to nk . Therefore picking δ to be half of what it was previously suffices for this matrix.

Note that this lower bound is strictly better than Theorem 17.1.1, since $n\sqrt{s/\delta} = \Omega(n/\delta)$ by $\delta = \Omega(1/s)$. It would be interesting to use this lower bound to interpolate between the lower bounds for matrix scaling and balancing in the high- and constant-precision regimes.

Bibliography

- [AAB+19] F. Arute, K. Arya, R. Babbush, et al. “Quantum Supremacy Using a Programmable Superconducting Processor”. In: *Nature* 574 (2019), pp. 505–510. doi: 10.1038/s41586-019-1666-5.
- [ABB+99] E. Anderson, Z. Bai, C. Bischof, et al. *LAPACK Users’ Guide*. SIAM, 1999.
- [ABM08] F. Alvarez, J. Bolte, and J. Munier. “A Unifying Local Convergence Result for Newton’s Method in Riemannian Manifolds”. In: *Foundations of Computational Mathematics* 8.2 (2008), pp. 197–226. doi: 10.1007/s10208-006-0221-6.
- [Afs11] B. Afsari. “Riemannian L^p Center of Mass: Existence, Uniqueness, and Convexity”. In: *Proceedings of the American Mathematical Society* 139.2 (2011), pp. 655–673. doi: 10.1090/S0002-9939-2010-10541-5.
- [AG19] J. van Apeldoorn and A. Gilyén. “Improvements in Quantum SDP-Solving with Applications”. In: *Proceedings of 46th International Colloquium on Automata, Languages, and Programming (ICALP)*. Vol. 132. Leibniz International Proceedings in Informatics (LIPIcs). 2019, 99:1–99:15. doi: 10.4230/LIPIcs.ICALP.2019.99.
- [AGGW20] J. van Apeldoorn, A. Gilyén, S. Gribling, and R. de Wolf. “Quantum SDP-Solvers: Better Upper and Lower Bounds”. In: *Quantum* 4.230 (2020).
- [AGL+18] Z. Allen-Zhu, A. Garg, Y. Li, R. Oliveira, and A. Wigderson. “Operator Scaling via Geodesically Convex Optimization, Invariant Theory and Polynomial Identity Testing”. In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. ACM, 2018, pp. 172–181. doi: 10.1145/3188745.3188942.
- [AGL+21] J. van Apeldoorn, S. Gribling, Y. Li, H. Nieuwboer, M. Walter, and R. de Wolf. “Quantum Algorithms for Matrix Scaling and Matrix Balancing”. In: *Proceedings of 48th International Colloquium on Automata, Languages, and Programming (ICALP)*. Vol. 198. 2021, 110:1–110:17. doi: 10.4230/LIPIcs.ICALP.2021.110.
- [AGN23] J. van Apeldoorn, S. Gribling, and H. Nieuwboer. *Basic Quantum Subroutines: Finding Multiple Marked Elements and Summing Numbers*. 2023. arXiv: 2302.10244.
- [AKRS21a] C. Améndola, K. Kohn, P. Reichenbach, and A. Seigal. “Invariant Theory and Scaling Algorithms for Maximum Likelihood Estimation”. In: *SIAM Journal on Applied Algebra and Geometry* 5.2 (2021), pp. 304–337. doi: 10.1137/20M1328932.
- [AKRS21b] C. Améndola, K. Kohn, P. Reichenbach, and A. Seigal. “Toric Invariant Theory for Maximum Likelihood Estimation in Log-Linear Models”. In: *Algebraic Statistics* 12.2 (2021), pp. 187–211. doi: 10.2140/astat.2021.12.187.
- [ALOW17] Z. Allen-Zhu, Y. Li, R. Oliveira, and A. Wigderson. “Much Faster Algorithms for Matrix Scaling”. In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. 2017, pp. 890–901. doi: 10.1109/FOCS.2017.87.
- [AM20] S. Arunachalam and R. Maity. “Quantum Boosting”. In: *Proceedings of 37th International Conference on Machine Learning (ICML)*. 2020.
- [Amb02] A. Ambainis. “Quantum Lower Bounds by Quantum Arguments”. In: *Journal of Computer and System Sciences* 64.4 (2002), pp. 750–767.
- [Amb07] A. Ambainis. “Quantum Walk Algorithm for Element Distinctness”. In: *SIAM Journal on Computing* 37.1 (2007), pp. 210–239. doi: 10.1137/S0097539705447311.
- [Amb10] A. Ambainis. *Variable Time Amplitude Amplification and a Faster Quantum Algorithm for Solving Systems of Linear Equations*. 2010. arXiv: 1010.4458.

- [AMN+23] A. Acuaviva, V. Makam, H. Nieuwboer, D. Pérez-García, F. Sittner, M. Walter, and F. Witteveen. “The Minimal Canonical Form of a Tensor Network”. In: *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*. 2023. arXiv: 2209.14358. Forthcoming.
- [AMRR11] A. Ambainis, L. Magnin, M. Roetteler, and J. Roland. “Symmetry-Assisted Adversaries for Quantum State Generation”. In: *2011 IEEE 26th Annual Conference on Computational Complexity (CCC)*. 2011, pp. 167–177. doi: 10.1109/CCC.2011.24. arXiv: 1012.2112.
- [AMS09] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009. doi: 10.1515/9781400830244.
- [AN13] M. Arnaudon and F. Nielsen. “On Approximating the Riemannian 1-Center”. In: *Computational Geometry* 46.1 (2013), pp. 93–104. doi: 10.1016/j.comgeo.2012.04.007.
- [ANR17] J. Altschuler, J. Niles-Weed, and P. Rigollet. “Near-Linear Time Approximation Algorithms for Optimal Transport via Sinkhorn Iteration”. In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017, pp. 1964–1974.
- [AP22] J. M. Altschuler and P. A. Parrilo. “Approximating Min-Mean-Cycle for Low-Diameter Graphs in Near-Optimal Time and Memory”. In: *SIAM Journal on Optimization* 32.3 (2022), pp. 1791–1816. doi: 10.1137/21M1439390.
- [AP23] J. M. Altschuler and P. A. Parrilo. “Near-Linear Convergence of the Random Osborne Algorithm for Matrix Balancing”. In: *Mathematical Programming* 198.1 (2023), pp. 363–397. doi: 10.1007/s10107-022-01825-4.
- [AR20] S. Aaronson and P. Rall. “Quantum Approximate Counting, Simplified”. In: *Symposium on Simplicity in Algorithms*. 2020, pp. 24–32. doi: 10.1137/1.9781611976014.5.
- [Art71] M. Artin. *Algebraic Spaces*. Yale University Press, 1971.
- [AS20] K. Ahn and S. Sra. “From Nesterov’s Estimate Sequence to Riemannian Acceleration”. In: *Proceedings of Thirty Third Conference on Learning Theory*. PMLR, 2020, pp. 84–118. URL: <https://proceedings.mlr.press/v125/ahn20a.html>.
- [Ati82] M. F. Atiyah. “Convexity and Commuting Hamiltonians”. In: *Bulletin of the London Mathematical Society* 14.1 (1982), pp. 1–15. doi: 10.1112/blms/14.1.1.
- [AV97] N. Alon and V. H. Vü. “Anti-Hadamard Matrices, Coin Weighing, Threshold Gates, and Indecomposable Hypergraphs”. In: *Journal of Combinatorial Theory, Series A* 79.1 (1997), pp. 133–160. doi: 10.1006/jcta.1997.2780.
- [AW17] S. Arunachalam and R. de Wolf. “Optimizing the Number of Gates in Quantum Search”. In: *Quantum Information and Computation* 17.1 (2017), pp. 251–261. doi: 10.26421/qic17.3-4. arXiv: 1512.07550.
- [AW22] S. Apers and R. de Wolf. “Quantum Speedup for Graph Sparsification, Cut Approximation, and Laplacian Solving”. In: *SIAM Journal on Computing* 51.6 (2022), pp. 1703–1742. doi: 10.1137/21M1391018.
- [AY98] E. D. Andersen and Y. Ye. “A Computational Study of the Homogeneous Algorithm for Large-Scale Convex Optimization”. In: *Computational Optimization and Applications* 10.3 (1998), pp. 243–269.
- [Ban38] S. Banach. “Über homogene Polynome in (L^2) ”. In: *Studia Mathematica* 7.1 (1938), pp. 36–44. URL: <https://eudml.org/doc/218624>.
- [BBBV97] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. “Strengths and Weaknesses of Quantum Computing”. In: *SIAM Journal on Computing* 26.5 (1997), pp. 1510–1523. doi: 10.1137/S0097539796300933.
- [BBC+01] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. “Quantum Lower Bounds by Polynomials”. In: *Journal of the ACM* 48.4 (2001), pp. 778–797. doi: 10.1145/502090.502097.
- [BBHT98] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. “Tight Bounds on Quantum Searching”. In: *Fortschritte der Physik* 46.4-5 (1998), pp. 493–505. arXiv: quant-ph/9605034.

- [BC13] P. Bürgisser and F. Cucker. *Condition: The Geometry of Numerical Algorithms*. Springer Science & Business Media, 2013.
- [BCK+23] J. van den Brand, L. Chen, R. Kyng, Y. P. Liu, R. Peng, M. P. Gutenberg, S. Sachdeva, and A. Sidford. “A Deterministic Almost-Linear Time Algorithm for Minimum-Cost Flow”. In: *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*. 2023. arXiv: 2309.16629. Forthcoming.
- [BCMw17] P. Bürgisser, M. Christandl, K. D. Mulmuley, and M. Walter. “Membership in Moment Polytopes is in NP and coNP”. In: *SIAM Journal on Computing* 46.3 (2017), pp. 972–991. doi: 10.1137/15M1048859.
- [BCS13] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. *Algebraic Complexity Theory*. Springer Science & Business Media, 2013.
- [BCSS98] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation*. Springer, 1998. doi: 10.1007/978-1-4612-0701-6.
- [BCWZ99] H. Buhrman, R. Cleve, R. de Wolf, and C. Zalka. “Bounds for Small-Error and Zero-Error Quantum Algorithms”. In: *40th Annual Symposium on Foundations of Computer Science*. IEEE Comput. Soc, 1999, pp. 358–368. doi: 10.1109/SFFCS.1999.814607.
- [BD00] R. Bhatia and C. Davis. “More Operator Versions of the Schwarz Inequality”. In: *Commun. Math. Phys.* 215 (2000), pp. 239–244.
- [BDM+21] P. Bürgisser, M. L. Doğan, V. Makam, M. Walter, and A. Wigderson. “Polynomial Time Algorithms in Invariant Theory for Torus Actions”. In: *36th Computational Complexity Conference (CCC)*. Vol. 200. Leibniz International Proceedings in Informatics (LIPIcs). 2021, 32:1–32:30. doi: 10.4230/LIPIcs.CCC.2021.32.
- [BDWY11] B. Barak, Z. Dvir, A. Wigderson, and A. Yehudayoff. “Rank Bounds for Design Matrices with Applications to Combinatorial Geometry and Locally Correctable Codes”. In: *Proceedings of 43rd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2011, pp. 519–528. doi: 10.1145/1993636.1993705.
- [BDWY12] B. Barak, Z. Dvir, A. Wigderson, and A. Yehudayoff. “Fractional Sylvester–Gallai Theorems”. In: *Proceedings of the National Academy of Sciences* (2012). doi: 10.1073/pnas.1203737109.
- [BE19] S. Bubeck and R. Eldan. “The Entropic Barrier: Exponential Families, Log-Concave Geometry, and Self-Concordance”. In: *Mathematics of Operations Research* 44.1 (2019), pp. 264–276. doi: 10.1287/moor.2017.0923.
- [Bel23] A. Belovs. *A Direct Reduction from the Polynomial to the Adversary Method*. 2023. arXiv: 2301.10317.
- [BFG+18] P. Bürgisser, C. Franks, A. Garg, R. Oliveira, M. Walter, and A. Wigderson. “Efficient Algorithms for Tensor Scaling, Quantum Marginals and Moment Polytopes”. In: *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. 2018, pp. 883–897. doi: 10.1109/FOCS.2018.00088. arXiv: 1804.04739.
- [BFG+19] P. Bürgisser, C. Franks, A. Garg, R. Oliveira, M. Walter, and A. Wigderson. “Towards a Theory of Non-Commutative Optimization: Geodesic 1st and 2nd Order Methods for Moment Maps and Polytopes”. In: *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. 2019, pp. 845–861. doi: 10.1109/FOCS.2019.00055.
- [BG19] N. Bansal and A. Gupta. “Potential-Function Proofs for Gradient Methods”. In: *Theory of Computing* 15.4 (4 2019), pp. 1–32. doi: 10.4086/toc.2019.v015a004.
- [BGO+18] P. Bürgisser, A. Garg, R. Oliveira, M. Walter, and A. Wigderson. “Alternating Minimization, Scaling Algorithms, and the Null-Cone Problem from Invariant Theory”. In: *9th Innovations in Theoretical Computer Science Conference (ITCS)*. Vol. 94. Leibniz International Proceedings in Informatics (LIPIcs). 2018, 24:1–24:20. doi: 10.4230/LIPIcs.ITCS.2018.24.
- [BH13] M. R. Bridson and A. Haefliger. *Metric Spaces of Non-Positive Curvature*. Vol. 319. Springer Science & Business Media, 2013.

- [Bha09] R. Bhatia. *Positive Definite Matrices*. Princeton University Press, 2009. doi: 10.1515/9781400827787.
- [Bha13] R. Bhatia. *Matrix Analysis*. Springer Science & Business Media, 2013. doi: 10.1007/978-1-4612-0653-8.
- [BHMT02] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. “Quantum Amplitude Amplification and Estimation”. In: *Quantum Computation and Quantum Information: A Millennium Volume*. Vol. 305. Contemporary Mathematics. American Mathematical Society, 2002, pp. 53–74.
- [BHS+21] R. Bergmann, R. Herzog, M. Silva Louzeiro, D. Tenbrinck, and J. Vidal-Núñez. “Fenchel Duality Theory and a Primal-Dual Algorithm on Riemannian Manifolds”. In: *Foundations of Computational Mathematics* 21.6 (2021), pp. 1465–1504. doi: 10.1007/s10208-020-09486-5.
- [BIL+21] M. Bläser, C. Ikenmeyer, V. Lysikov, A. Pandey, and F.-O. Schreyer. “On the Orbit Closure Containment Problem and Slice Rank of Tensors”. In: *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2021, pp. 2565–2584. doi: 10.1137/1.9781611976465.152.
- [BIP19] P. Bürgisser, C. Ikenmeyer, and G. Panova. “No Occurrence Obstructions in Geometric Complexity Theory”. In: *Journal of the American Mathematical Society* 32.1 (2019), pp. 163–193. doi: 10.1090/jams/908.
- [Bir71] D. Birkes. “Orbits of Linear Algebraic Groups”. In: *Annals of Mathematics* 93.3 (1971), pp. 459–475. doi: 10.2307/1970884.
- [BKL+19] F. Brandão, A. Kalev, T. Li, C. Y.-Y. Lin, K. Svore, and X. Wu. “Quantum SDP Solvers: Large Speed-Ups, Optimality, and Applications to Quantum Learning”. In: *Proceedings of 46th International Colloquium on Automata, Languages, and Programming (ICALP)*. Vol. 132. Leibniz International Proceedings in Informatics (LIPIcs). 2019, 27:1–27:14. doi: 10.4230/LIPIcs.ICALP.2019.27.
- [BKVH07] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi. “A Tutorial on Geometric Programming”. In: *Optimization and Engineering* 8.1 (2007), p. 67. doi: 10.1007/s11081-007-9001-7.
- [BLMW11] P. Bürgisser, J. M. Landsberg, L. Manivel, and J. Weyman. “An Overview of Mathematical Issues Arising in the Geometric Complexity Theory Approach to $VP \neq VNP$ ”. In: *SIAM Journal on Computing* 40.4 (2011), pp. 1179–1209. doi: 10.1137/090765328.
- [BLNW20] P. Bürgisser, Y. Li, H. Nieuwboer, and M. Walter. *Interior-Point Methods for Unconstrained Geometric Programming and Scaling Problems*. 2020. arXiv: 2008.12110.
- [Bou23] N. Boumal. *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 2023. doi: 10.1017/9781009166164.
- [Bra10] A. M. Bradley. “Algorithms for the Equilibration of Matrices and Their Application to Limited-Memory Quasi-Newton Methods”. PhD thesis. Stanford University, 2010.
- [Bra19] J. van den Brand. “A Deterministic Linear Program Solver in Current Matrix Multiplication Time”. In: *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics, 2019, pp. 259–278. doi: 10.1137/1.9781611975994.16.
- [Bri87] M. Brion. “Sur l’image de l’application moment”. In: *Séminaire d’Algèbre Paul Dubreil et Marie-Paule Malliavin*. Ed. by M.-P. Malliavin. Lecture Notes in Mathematics. Springer, 1987, pp. 177–192. doi: 10.1007/BFb0078526.
- [Bri88] M. Brion. “Points entiers dans les polyèdres convexes”. In: *Annales scientifiques de l’École normale supérieure* 21.4 (1988), pp. 653–663. doi: 10.24033/asens.1572.
- [BRV18] J. Bryan, Z. Reichstein, and M. Van Raamsdonk. “Existence of Locally Maximally Entangled Quantum States via Geometric Invariant Theory”. In: *Annales Henri Poincaré* 19.8 (2018), pp. 2491–2511.

- [BS17] F. Brandão and K. Svore. “Quantum Speed-Ups for Solving Semidefinite Programs”. In: *Proceedings of 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*. 2017, pp. 415–426. doi: 10.1109/FOCS.2017.45.
- [BV04] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge: Cambridge University Press, 2004.
- [BZ11] R. P. Brent and P. Zimmermann. *Modern Computer Arithmetic*. Vol. 18. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 2011, pp. xvi+221.
- [Car52] É. Cartan. “La théorie des groupes finis et continus et l’analysis situs”. In: (1952). URL: <https://eudml.org/doc/192641>.
- [Car72] B. C. Carlson. “The Logarithmic Mean”. In: *The American Mathematical Monthly* 79.6 (1972), pp. 615–618. doi: 10.1080/00029890.1972.11993095.
- [CB22] C. Criscitiello and N. Boumal. “Negative Curvature Obstructs Acceleration for Strongly Geodesically Convex Optimization, Even with Exact First-Order Oracles”. In: *Proceedings of Thirty Fifth Conference on Learning Theory*. PMLR, 2022, pp. 496–542. URL: <https://proceedings.mlr.press/v178/criscitiello22a.html>.
- [CB23] C. Criscitiello and N. Boumal. “Curvature and Complexity: Better Lower Bounds for Geodesically Convex Optimization”. In: *Proceedings of Thirty Sixth Conference on Learning Theory*. PMLR, 2023, pp. 2969–3013. URL: <https://proceedings.mlr.press/v195/criscitiello23a.html>.
- [CGFW21] M. Christandl, F. Gessmundo, D. S. França, and A. H. Werner. “Optimization at the Boundary of the Tensor Network Variety”. In: *Physical Review B* 103.19 (2021), p. 195139.
- [CGQ+23] Z. Chen, J. A. Grochow, Y. Qiao, G. Tang, and C. Zhang. *On the Complexity of Isomorphism Problems for Tensors, Groups, and Polynomials III: Actions by Classical Groups*. 2023. arXiv: 2306.03135.
- [CGW11] X. Chen, Z.-C. Gu, and X.-G. Wen. “Classification of Gapped Symmetric Phases in 1D Spin Systems”. In: *PRB* 83 (2011), p. 035107. eprint: arXiv:1008.3745.
- [Che23] S. Chewi. “The Entropic Barrier is n-Self-Concordant”. In: *Geometric Aspects of Functional Analysis*. Ed. by R. Eldan, B. Klartag, A. Litvak, and E. Milman. Lecture Notes in Mathematics. Cham: Springer International Publishing, 2023, pp. 209–222. doi: 10.1007/978-3-031-26300-2_6.
- [CHI+18] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig. “Quantum Machine Learning: A Classical Perspective”. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 474.20170551 (2018). doi: 10.1098/rspa.2017.0551.
- [Chi22] A. Childs. *Lecture Notes on Quantum Algorithms*. 2022.
- [CK21] D. Chakrabarty and S. Khanna. “Better and Simpler Error Analysis of the Sinkhorn–Knopp Algorithm for Matrix Scaling”. In: *Mathematical Programming* 188.1 (2021), pp. 395–407. doi: 10.1007/s10107-020-01503-3.
- [CKL+22] L. Chen, R. Kyng, Y. P. Liu, R. Peng, M. P. Gutenberg, and S. Sachdeva. “Maximum Flow and Minimum-Cost Flow in Almost-Linear Time”. In: *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. 2022, pp. 612–623. doi: 10.1109/FOCS54457.2022.00064.
- [CKV20] L. E. Celis, V. Keswani, and N. Vishnoi. “Data Preprocessing to Mitigate Bias: A Maximum Entropy Based Approach”. In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020, pp. 1349–1359. URL: <https://proceedings.mlr.press/v119/celis20a.html>.
- [CLM+16] M. B. Cohen, Y. T. Lee, G. Miller, J. Pachocki, and A. Sidford. “Geometric Median in Nearly Linear Time”. In: *Proceedings of the 48th Annual ACM Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2016, pp. 9–21. doi: 10.1145/2897518.2897647.

- [CLRS22] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Fourth Edition*. MIT Press, 2022.
- [CLVW20] M. Christandl, A. Lucia, P. Vrana, and A. H. Werner. “Tensor Network Representations from the Geometry of Entangled States”. In: *SciPost Physics* 9.3 (2020), p. 042.
- [CM06] M. Christandl and G. Mitchison. “The Spectra of Quantum States and the Kronecker Coefficients of the Symmetric Group”. In: *Communications in Mathematical physics* 261.3 (2006), pp. 789–797.
- [CMB23] C. Criscitiello, D. Martínez-Rubio, and N. Boumal. “Open Problem: Polynomial Linearly-Convergent Method for g -Convex Optimization?” In: *Proceedings of Thirty Sixth Conference on Learning Theory*. PMLR, 2023, pp. 5950–5956. URL: <https://proceedings.mlr.press/v195/criscitiello23b.html>.
- [CMTV17] M. B. Cohen, A. Mađry, D. Tsipras, and A. Vladu. “Matrix Scaling and Balancing via Box Constrained Newton’s Method and Interior Point Methods”. In: *Proceedings of 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*. 2017, pp. 902–913. DOI: 10.1109/FOCS.2017.88.
- [CPSV11] J. I. Cirac, D. Poilblanc, N. Schuch, and F. Verstraete. “Entanglement Spectrum and Boundary Theories with Projected Entangled-Pair States”. In: *Physical Review B* 83.24 (2011), p. 245134.
- [CPSV17] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete. “Matrix Product Density Operators: Renormalization Fixed Points and Boundary Theories”. In: *Annals of Physics* 378 (2017), pp. 100–149. DOI: 10.1016/j.aop.2016.12.030.
- [CPSV21] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete. “Matrix Product States and Projected Entangled Pair States: Concepts, Symmetries, and Theorems”. In: *Reviews of Modern Physics* 93.4 (2021), p. 045003. DOI: 10.1103/RevModPhys.93.045003.
- [CPZ+17] A. Cichocki, A.-H. Phan, Q. Zhao, N. Lee, I. Oseledets, M. Sugiyama, D. P. Mandic, et al. “Tensor Networks for Dimensionality Reduction and Large-Scale Optimization: Part 2 Applications and Future Perspectives”. In: *Foundations and Trends in Machine Learning* 9.6 (2017), pp. 431–673.
- [Cut13] M. Cuturi. “Sinkhorn Distances: Lightspeed Computation of Optimal Transport”. In: *Advances in Neural Information Processing Systems*. Vol. 26. Curran Associates, Inc., 2013. URL: <https://proceedings.neurips.cc/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html>.
- [CVZ21] M. Christandl, P. Vrana, and J. Zuiddam. “Universal Points in the Asymptotic Spectrum of Tensors”. In: *Journal of the American Mathematical Society* 36.1 (2021), pp. 31–79. DOI: 10.1090/jams/996.
- [Dan63] G. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1963. DOI: 10.1515/9781400884179.
- [DCS18] H. Dreyer, J. I. Cirac, and N. Schuch. “Projected Entangled Pair States with Continuous Virtual Symmetries”. In: *Physical Review B* 98.11 (2018), p. 115120.
- [DCSP17] G. De las Cuevas, J. I. Cirac, N. Schuch, and D. Pérez-García. “Irreducible Forms of Matrix Product States: Theory and Applications”. In: *Journal of Mathematical Physics* 58.12 (2017), p. 121901. DOI: 10.1063/1.5000784.
- [Der00] H. Derksen. “Polynomial Bounds for Rings of Invariants”. In: *Proceedings of the American Mathematical Society* 129.4 (2000), pp. 955–963. DOI: 10.1090/S0002-9939-00-05698-7.
- [Der22] H. Derksen. “The G -stable Rank for Tensors and the Cap Set Problem”. In: *Algebra & Number Theory* 16.5 (2022), pp. 1071–1097. DOI: 10.2140/ant.2022.16.1071.
- [Deu85] D. Deutsch. “Quantum Theory, the Church–Turing Principle and the Universal Quantum Computer”. In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400.1818 (1985), pp. 97–117. DOI: 10.1098/rspa.1985.0070.

- [Deu89] D. E. Deutsch. “Quantum Computational Networks”. In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 425.1868 (1989), pp. 73–90. doi: 10.1098/rspa.1989.0099.
- [DGK18] P. Dvurechensky, A. Gasnikov, and A. Kroshnin. “Computational Optimal Transport: Complexity by Accelerated Gradient Descent Is Better Than by Sinkhorn’s Algorithm”. In: *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2018, pp. 1367–1376. URL: <https://proceedings.mlr.press/v80/dvurechensky18a.html>.
- [DGOS18] Z. Dvir, A. Garg, R. Oliveira, and J. Solymosi. “Rank bounds for design matrices with block entries and geometric applications”. In: *Discrete Analysis* 5.2018 (2018), pp. 1–24. doi: 10.19086/da.3118.
- [DH05] S. Daftuar and P. Hayden. “Quantum State Transformations and the Schubert Calculus”. In: *Annals of Physics* 315.1 (2005), pp. 80–122.
- [DHHM06] C. Dürr, M. Heiligman, P. Høyer, and M. Mhalla. “Quantum Query Complexity of Some Graph Problems”. In: *SIAM Journal on Computing* 35.6 (2006), pp. 1310–1328. doi: 10.1137/050644719.
- [DIP20] J. Dörfler, C. Ikenmeyer, and G. Panova. “On Geometric Complexity Theory: Multiplicity Obstructions Are Stronger Than Occurrence Obstructions”. In: *SIAM Journal on Applied Algebra and Geometry* (2020). doi: 10.1137/19M1287638.
- [DK15] H. Derksen and G. Kemper. *Computational Invariant Theory*. 2nd ed. 2015. Encyclopaedia of Mathematical Sciences. Springer Berlin Heidelberg, 2015. doi: 10.1007/978-3-662-48422-7.
- [DKMV23] H. Derksen, I. Klep, V. Makam, and J. Volčič. “Ranks of Linear Matrix Pencils Separate Simultaneous Similarity Orbits”. In: *Advances in Mathematics* 415 (2023), p. 108888. doi: 10.1016/j.aim.2023.108888.
- [DM20a] H. Derksen and V. Makam. “Algorithms for Orbit Closure Separation for Invariants and Semi-Invariants of Matrices”. In: *Algebra & Number Theory* 14.10 (2020), pp. 2791–2813. doi: 10.2140/ant.2020.14.2791.
- [DM20b] H. Derksen and V. Makam. “An Exponential Lower Bound for the Degrees of Invariants of Cubic Forms and Tensor Actions”. In: *Advances in Mathematics* 368 (2020), p. 107136. doi: 10.1016/j.aim.2020.107136.
- [DM21] H. Derksen and V. Makam. “Maximum Likelihood Estimation for Matrix Normal Models via Quiver Representations”. In: *SIAM Journal on Applied Algebra and Geometry* 5.2 (2021), pp. 338–365. doi: 10.1137/20M1369348.
- [DM22] H. Derksen and V. Makam. “Polystability in Positive Characteristic and Degree Lower Bounds for Invariant Rings”. In: *Journal of Combinatorial Algebra* 6.3 (2022), pp. 353–405. doi: 10.4171/jca/66.
- [DMW22] H. Derksen, V. Makam, and M. Walter. “Maximum Likelihood Estimation for Tensor Normal Models via Castling Transforms”. In: *Forum of Mathematics, Sigma* 10 (2022), e50. doi: 10.1017/fms.2022.37.
- [DP15] A. Dolcetti and D. Pertici. “Some Differential Properties of $GL_n(\mathbb{R})$ with the Trace Metric”. In: *Rivista di Matematica della Università di Parma* 6 (2015), pp. 267–286. arXiv: 1412.4565.
- [DP17] C. De Concini and C. Procesi. *The Invariant Theory of Matrices*. Vol. 69. University Lecture Series. American Mathematical Society, 2017. doi: 10.1090/ulect/069.
- [DPM03] J.-P. Dedieu, P. Priouret, and G. Malajovich. “Newton’s Method on Riemannian Manifolds: Covariant Alpha Theory”. In: *IMA Journal of Numerical Analysis* 23.3 (2003), pp. 395–419. doi: 10.1093/imanum/23.3.395.
- [DPZ67] R. J. Duffin, E. L. Peterson, and C. Zener. *Geometric Programming – Theory and Application*. John Wiley & Sons, 1967.
- [DSW14] Z. Dvir, S. Saraf, and A. Wigderson. “Improved Rank Bounds for Design Matrices and a New Proof of Kelly’s Theorem”. In: *Forum of Mathematics, Sigma* 2 (2014). doi: 10.1017/fms.2014.2.

- [Dui99] J. J. Duistermaat. *On the Boundary Behaviour of the Riemannian Structure of a Self-Concordant Barrier Function*. 1999.
- [DVZ20] D. Dadush, L. A. Végh, and G. Zambelli. “Rescaling Algorithms for Linear Conic Feasibility”. In: *Mathematics of Operations Research* 45.2 (2020), pp. 732–754. doi: 10.1287/moor.2019.1011.
- [Ebe97] P. B. Eberlein. *Geometry of Nonpositively Curved Manifolds*. Chicago Lectures in Mathematics. University of Chicago Press, 1997.
- [Eve18] G. Evenbly. “Gauge Fixing, Canonical Forms and Optimal Truncations in Tensor Networks with Closed Loops”. In: *Physical Review B* 98.8 (2018), p. 085155. doi: 10.1103/PhysRevB.98.085155.
- [FNW92] M. Fannes, B. Nachtergaele, and R. F. Werner. “Finitely Correlated States on Quantum Spin Chains”. In: *Commun. Math. Phys.* 144 (1992), p. 443.
- [For01] J. Forster. “A Linear Lower Bound on the Unbounded Error Probabilistic Communication Complexity”. In: *Proceedings of 16th IEEE Annual Conference on Computational Complexity*. 2001, pp. 100–106. doi: 10.1109/CCC.2001.933877.
- [For86] E. Formanek. “Generating the Ring of Matrix Invariants”. In: *Ring Theory*. Ed. by F. M. J. van Oystaeyen. Lecture Notes in Mathematics. Springer, 1986, pp. 73–82. doi: 10.1007/BFb0076314.
- [FORW21] C. Franks, R. Oliveira, A. Ramachandran, and M. Walter. *Near Optimal Sample Complexity for Matrix and Tensor Normal Models via Geodesic Convexity*. 2021. arXiv: 2110.07583.
- [Fox15] D. J. F. Fox. “A Schwarz Lemma for Kähler Affine Metrics and the Canonical Potential of a Proper Convex Cone”. In: *Annali di Matematica Pura ed Applicata (1923 -)* 194.1 (2015), pp. 1–42. doi: 10.1007/s10231-013-0362-6.
- [FR21] C. Franks and P. Reichenbach. “Barriers for Recent Methods in Geodesic Optimization”. In: *36th Computational Complexity Conference (CCC)*. Vol. 200. Leibniz International Proceedings in Informatics (LIPIcs). 2021, 13:1–13:54. doi: 10.4230/LIPIcs.CCC.2021.13.
- [Fra18] C. Franks. “Operator Scaling with Specified Marginals”. In: *Proceedings of the 50th ACM Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2018, pp. 190–203. doi: 10.1145/3188745.3188932.
- [Fri55] K. R. Frisch. “The Logarithmic Potential Method of Convex Programming”. In: *Memorandum, University Institute of Economics, Oslo* 5.6 (1955).
- [FS02] O. P. Ferreira and B. F. Svaiter. “Kantorovich’s Theorem on Newton’s Method in Riemannian Manifolds”. In: *Journal of Complexity* 18.1 (2002), pp. 304–329. doi: 10.1006/jcom.2001.0582.
- [FS22] H. Fawzi and J. Saunderson. *Optimal Self-Concordant Barriers for Quantum Relative Entropies*. 2022. arXiv: 2205.04581.
- [FSG23] C. Franks, T. Soma, and M. X. Goemans. “Shrunk Subspaces via Operator Sinkhorn Iteration”. In: *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Proceedings. Society for Industrial and Applied Mathematics, 2023, pp. 1655–1668. doi: 10.1137/1.9781611977554.ch62.
- [FVJ09] P. T. Fletcher, S. Venkatasubramanian, and S. Joshi. “The Geometric Median on Riemannian Manifolds with Application to Robust Atlas Estimation”. In: *NeuroImage* 45 (2009), S143–S152. doi: 10.1016/j.neuroimage.2008.10.052. pmid: 19056498.
- [FW20] C. Franks and M. Walter. *Minimal Length in an Orbit Closure as a Semiclassical Limit*. 2020. arXiv: 2004.14872.
- [FZ20] L. Faybusovich and C. Zhou. “Self-Concordance and Matrix Monotonicity with Applications to Quantum Entanglement Problems”. In: *Applied Mathematics and Computation* 375 (2020), p. 125071. doi: 10.1016/j.amc.2020.125071.

- [GGOW16] A. Garg, L. Gurvits, R. Oliveira, and A. Wigderson. “A Deterministic Polynomial Time Algorithm for Non-commutative Rational Identity Testing”. In: *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2016, pp. 109–117. doi: 10.1109/FOCS.2016.95.
- [GGOW18] A. Garg, L. Gurvits, R. Oliveira, and A. Wigderson. “Algorithmic and Optimization Aspects of Brascamp-Lieb Inequalities, via Operator Scaling”. In: *Geometric and Functional Analysis* 28.1 (2018), pp. 100–145. doi: 10.1007/s00039-018-0434-2.
- [GGOW20] A. Garg, L. Gurvits, R. Oliveira, and A. Wigderson. “Operator Scaling: Theory and Applications”. In: *Foundations of Computational Mathematics* 20.2 (2020), pp. 223–290. doi: 10.1007/s10208-019-09417-z.
- [GH78] P. Griffiths and J. Harris. *Principles of Algebraic Geometry*. John Wiley & Sons, 1978.
- [GIM+20] A. Garg, C. Ikenmeyer, V. Makam, R. Oliveira, M. Walter, and A. Wigderson. “Search Problems in Algebraic Complexity, GCT, and Hardness of Generators for Invariant Rings”. In: *Proceedings of the 35th Computational Complexity Conference (CCC)*. 2020, pp. 1–17. doi: 10.4230/LIPIcs.CCC.2020.12.
- [GK72] Yu. Sh. Gurevich and I. O. Koryakov. “Remarks on Berger’s Paper on the Domino Problem”. In: *Siberian Mathematical Journal* 13.2 (1972), pp. 319–321. doi: 10.1007/BF00971620.
- [GLM08] V. Giovannetti, S. Lloyd, and L. Maccone. “Quantum Random Access Memory”. In: *Physical Review Letters* 100.16 (2008), p. 160501. doi: 10.1103/PhysRevLett.100.160501.
- [GLS12] M. Grötschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer Science & Business Media, 2012.
- [GMS+86] P. E. Gill, W. Murray, M. A. Saunders, J. A. Tomlin, and M. H. Wright. “On Projected Newton Barrier Methods for Linear Programming and an Equivalence to Karmarkar’s Projective Method”. In: *Mathematical Programming* 36.2 (1986), pp. 183–209. doi: 10.1007/BF02592025.
- [GN22] S. Gribling and H. Nieuwboer. “Improved Quantum Lower and Upper Bounds for Matrix Scaling”. In: *Proceedings of 39th International Symposium on Theoretical Aspects of Computer Science (STACS)*. Vol. 219. 2022, 35:1–35:23. doi: 10.4230/LIPIcs.STACS.2022.35.
- [GN99] X. Gual-Arnau and A. M. Naveira. “Volume of Tubes in Noncompact Symmetric Spaces”. In: *Publ. Math. Debrecen* 54.3-4 (1999), pp. 313–320.
- [GO18] A. Garg and R. Oliveira. “Recent Progress on Scaling Algorithms and Applications”. In: *Bulletin of the EATCS, Computational Complexity Column* 125 (2018).
- [Gof80] J. L. Goffin. “The Relaxation Method for Solving Systems of Linear Inequalities”. In: *Mathematics of Operations Research* 5.3 (1980), pp. 388–414. doi: 10.1287/moor.5.3.388.
- [Gro96] L. K. Grover. “A Fast Quantum Mechanical Algorithm for Database Search”. In: *Proceedings of the 28th ACM Symposium on Theory of Computing (STOC)*. ACM Press, 1996, pp. 212–219. doi: 10.1145/237814.237866.
- [Gro97] L. K. Grover. *Quantum Telecomputation*. 1997. arXiv: quant-ph/9704012.
- [Gro98] L. K. Grover. “A Framework for Fast Quantum Mechanical Algorithms”. In: *Proceedings of the 30th Annual ACM Symposium on Theory of Computing (STOC)*. ACM Press, 1998, pp. 53–62. doi: 10.1145/276698.276712.
- [GS82] V. Guillemin and S. Sternberg. “Convexity Properties of the Moment Mapping”. In: *Inventiones mathematicae* 67.3 (1982), pp. 491–513. doi: 10.1007/BF01398933.
- [GS84] V. Guillemin and S. Sternberg. “Convexity Properties of the Moment Mapping. II”. In: *Inventiones mathematicae* 77.3 (1984), pp. 533–546. doi: 10.1007/BF01388837.
- [GSLW19] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe. “Quantum Singular Value Transformation and beyond: Exponential Improvements for Quantum Matrix Arithmetics”. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2019, pp. 193–204. doi: 10.1145/3313276.3316366.

- [Gur04] L. Gurvits. “Classical Complexity and Quantum Entanglement”. In: *Journal of Computer and System Sciences* 69.3 (2004), pp. 448–484. doi: 10.1016/j.jcss.2004.06.003.
- [GW02] M. de Graaf and R. de Wolf. “On Quantum Versions of the Yao Principle”. In: *STACS 2002. Lecture Notes in Computer Science*. Springer, 2002, pp. 347–358. doi: 10.1007/3-540-45841-7_28.
- [GW09] R. Goodman and N. R. Wallach. *Symmetry, Representations, and Invariants*. 1st ed. Graduate Texts in Mathematics 255. Springer, 2009.
- [GW10] G. Gour and N. R. Wallach. “All Maximally Entangled Four Qubits States”. In: *Journal of Mathematical Physics* 51.11 (2010), p. 112201. doi: 10.1063/1.3511477.
- [Ham21] Y. Hamoudi. “Quantum Sub-Gaussian Mean Estimator”. In: *29th Annual European Symposium on Algorithms (ESA)*. Vol. 204. Leibniz International Proceedings in Informatics (LIPIcs). 2021, 50:1–50:17. doi: 10.4230/LIPIcs.ESA.2021.50.
- [Har77] R. Hartshorne. *Algebraic Geometry*. Vol. 52. Graduate Texts in Mathematics. Springer, 1977. doi: 10.1007/978-1-4757-3849-0.
- [Hel79] S. Helgason. *Differential Geometry, Lie Groups, and Symmetric Spaces*. Academic Press, 1979.
- [HH21] M. Hamada and H. Hirai. “Computing the Nc-Rank via Discrete Convex Optimization on CAT(0) Spaces”. In: *SIAM Journal on Applied Algebra and Geometry* (2021). doi: 10.1137/20M138836X.
- [HHL09] A. W. Harrow, A. Hassidim, and S. Lloyd. “Quantum Algorithm for Linear Systems of Equations”. In: *Physical Review Letters* 103.15 (2009), p. 150502. doi: 10.1103/PhysRevLett.103.150502.
- [Hil14] R. Hildebrand. “Canonical Barriers on Convex Cones”. In: *Mathematics of Operations Research* 39.3 (2014), pp. 841–850. doi: 10.1287/moor.2013.0640.
- [Hil93] D. Hilbert. “Über die vollen Invariantensysteme”. In: *Mathematische Annalen* 42 (1893), pp. 313–373. url: <https://eudml.org/doc/urn:eudml:doc:157652>.
- [Hir22a] H. Hirai. *Convex Analysis on Hadamard Spaces and Scaling Problems*. 2022. arXiv: 2203.03193.
- [Hir22b] H. Hirai. *On a Manifold Formulation of Self-Concordant Functions*. 2022. arXiv: 2212.10981.
- [HLŠ07] P. Høyer, T. Lee, and R. Špalek. “Negative Weights Make Adversaries Stronger”. In: *Proceedings of the 39th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. 2007, pp. 526–535. doi: 10.1145/1250790.1250867.
- [HM21a] L. Hamilton and A. Moitra. “A No-Go Theorem for Robust Acceleration in the Hyperbolic Plane”. In: *Advances in Neural Information Processing Systems*. Vol. 34. Curran Associates, Inc., 2021, pp. 3914–3924. url: https://proceedings.neurips.cc/paper_files/paper/2021/hash/201d546992726352471cfea6b0df0a48-Abstract.html.
- [HM21b] L. Hamilton and A. Moitra. “The Paulsen Problem Made Simple”. In: *Israel Journal of Mathematics* 246.1 (2021), pp. 299–313. doi: 10.1007/s11856-021-2245-7.
- [HNQ+16] P. Hayden, S. Nezami, X.-L. Qi, N. Thomas, M. Walter, and Z. Yang. “Holographic Duality from Random Tensor Networks”. In: *Journal of High Energy Physics* 2016.11 (2016), pp. 1–56.
- [HNW23] H. Hirai, H. Nieuwboer, and M. Walter. “Interior-Point Methods on Manifolds: Theory and Applications”. In: *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*. 2023. arXiv: 2303.04771. Forthcoming.
- [HV17] J. Haegeman and F. Verstraete. “Diagonalizing Transfer Matrices and Matrix Product Operators: A Medley of Exact and Computational Methods”. In: *Annual Review of Condensed Matter Physics* 8 (2017), pp. 355–406.
- [Ide16] M. Idel. *A Review of Matrix Scaling and Sinkhorn’s Normal Form for Matrices and Positive Maps*. 2016. arXiv: 1609.06349.

- [IdW23] A. Izdebski and R. de Wolf. “Improved Quantum Boosting”. In: *31st Annual European Symposium on Algorithms (ESA)*. Vol. 274. Leibniz International Proceedings in Informatics (LIPIcs). 2023, 64:1–64:16. doi: 10.4230/LIPIcs.ESA.2023.64.
- [IMW17] C. Ikenmeyer, K. D. Mulmuley, and M. Walter. “On Vanishing of Kronecker Coefficients”. In: *computational complexity* 26.4 (2017), pp. 949–992. doi: 10.1007/s00037-017-0158-y.
- [Inn82] N. Innami. “Splitting Theorems of Riemannian Manifolds”. In: *Compositio Mathematica* 47.3 (1982), pp. 237–247. URL: http://www.numdam.org/item/CM_1982__47_3_237_0/.
- [IP17] C. Ikenmeyer and G. Panova. “Rectangular Kronecker Coefficients and Plethysms in Geometric Complexity Theory”. In: *Advances in Mathematics* 319 (2017), pp. 40–66. doi: 10.1016/j.aim.2017.08.024.
- [IQS17] G. Ivanyos, Y. Qiao, and K. V. Subrahmanyam. “Non-Commutative Edmonds’ Problem and Matrix Semi-Invariants”. In: *computational complexity* 26.3 (2017), pp. 717–763. doi: 10.1007/s00037-016-0143-x.
- [IQS18] G. Ivanyos, Y. Qiao, and K. V. Subrahmanyam. “Constructive Non-Commutative Rank Computation Is in Deterministic Polynomial Time”. In: *computational complexity* 27.4 (2018), pp. 561–593. doi: 10.1007/s00037-018-0165-7.
- [Jan18] S. Janson. “Tail Bounds for Sums of Geometric and Exponential Variables”. In: *Statistics & Probability Letters* 135 (2018), pp. 1–6. doi: 10.1016/j.spl.2017.11.017.
- [Jay57a] E. T. Jaynes. “Information Theory and Statistical Mechanics”. In: *Phys. Rev.* 106.4 (1957), pp. 620–630. doi: 10.1103/PhysRev.106.620.
- [Jay57b] E. T. Jaynes. “Information Theory and Statistical Mechanics. II”. In: *Phys. Rev.* 108.2 (1957), pp. 171–190. doi: 10.1103/PhysRev.108.171.
- [JCF+21] J. L. Jiménez, S. P. G. Crone, E. Fogh, et al. “A Quantum Magnetic Analogue to the Critical Point of Water”. In: *Nature* 592.7854 (7854 2021), pp. 370–375. doi: 10.1038/s41586-021-03411-8.
- [Ji07] H. Ji. “Optimization Approaches on Smooth Manifolds”. PhD thesis. Australian National University, 2007.
- [JM07] D. Jiang, J. B. Moore, and H. Ji. “Self-Concordant Functions for Optimization on Smooth Manifolds”. In: *Journal of Global Optimization* 38.3 (2007), pp. 437–457. doi: 10.1007/s10898-006-9095-z.
- [JR21] E. Jeandel and M. Rao. “An Aperiodic Set of 11 Wang Tiles”. In: *Advances in Combinatorics* 1 (2021). doi: 10.19086/aic.18614.
- [JWX08] H.-C. Jiang, Z.-Y. Weng, and T. Xiang. “Accurate Determination of Tensor Network State of Quantum Lattice Models in Two Dimensions”. In: *Physical Review Letters* 101.9 (2008), p. 090603.
- [Kar84a] N. Karmarkar. “A New Polynomial-Time Algorithm for Linear Programming”. In: *Proceedings of the 16th Annual ACM Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 1984, pp. 302–311. doi: 10.1145/800057.808695.
- [Kar84b] N. Karmarkar. “A New Polynomial-Time Algorithm for Linear Programming”. In: *Combinatorica* 4.4 (1984), pp. 373–395. doi: 10.1007/BF02579150.
- [Kem78] G. R. Kempf. “Instability in Invariant Theory”. In: *Annals of Mathematics* 108.2 (1978), pp. 299–316. doi: 10.2307/1971168.
- [Kha80] L. G. Khachiyan. “Polynomial Algorithms in Linear Programming”. In: *USSR Computational Mathematics and Mathematical Physics* 20.1 (1980), pp. 53–72. doi: 10.1016/0041-5553(80)90061-0.
- [Kir84a] F. Kirwan. “Convexity Properties of the Moment Mapping, III”. In: *Inventiones mathematicae* 77.3 (1984), pp. 547–552. doi: 10.1007/BF01388838.

- [Kir84b] F. C. Kirwan. *Cohomology of Quotients in Symplectic and Algebraic Geometry*. Vol. 31. Mathematical Notes. Princeton University Press, 1984. URL: <http://www.jstor.org/stable/j.ctv10vm2m8>.
- [Kir98] F. Kirwan. “Momentum Maps and Reduction in Algebraic Geometry”. In: *Differential Geometry and its Applications*. Symplectic Geometry 9.1 (1998), pp. 135–171. DOI: 10.1016/S0926-2245(98)00020-5.
- [KK93] B. Kalantari and L. Khachiyan. “On the Rate of Convergence of Deterministic and Randomized RAS Matrix Scaling Algorithms”. In: *Operations Research Letters* 14.5 (1993), pp. 237–244. DOI: 10.1016/0167-6377(93)90087-W.
- [KK96] B. Kalantari and L. Khachiyan. “On the Complexity of Nonnegative-Matrix Scaling”. In: *Linear Algebra and its Applications* 240 (1996), pp. 87–103. DOI: 10.1016/0024-3795(94)00188-X.
- [KKOS12] H. Kalis, D. Klagges, R. Orús, and K. P. Schmidt. “Fate of the Cluster State on the Square Lattice in a Magnetic Field”. In: *Physical Review A* 86.2 (2012), p. 022317.
- [KLLR18] T. C. Kwok, L. C. Lau, Y. T. Lee, and A. Ramachandran. “The Paulsen Problem, Continuous Operator Scaling, and Smoothed Analysis”. In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2018, pp. 182–189. DOI: 10.1145/3188745.3188794.
- [KLP+16] R. Kyng, Y. T. Lee, R. Peng, S. Sachdeva, and D. Spielman. “Sparsified Cholesky and Multigrid Solvers for Connection Laplacians”. In: *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. 2016, pp. 842–850. DOI: 10.1145/2897518.2897640.
- [KLP19] M. J. Kastoryano, A. Lucia, and D. Pérez-García. “Locality at the Boundary Implies Gap in the Bulk for 2D PEPS”. In: *Communications in Mathematical Physics* 366.3 (2019), pp. 895–926.
- [KLR19] T. C. Kwok, L. C. Lau, and A. Ramachandran. “Spectral Analysis of Matrix Scaling and Operator Scaling”. In: *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. 2019, pp. 1184–1204. DOI: 10.1109/FOCS.2019.00074.
- [KLRS07] B. Kalantari, I. Lari, F. Ricca, and B. Simeone. “On the Complexity of General Matrix Scaling and Entropy Minimization via the RAS Algorithm”. In: *Mathematical Programming* 112.2 (2007), pp. 371–401. DOI: 10.1007/s10107-006-0021-4.
- [Kly02] A. Klyachko. *Coherent States, Entanglement, and Geometric Invariant Theory*. 2002. arXiv: quant-ph/0206012.
- [Kly04] A. Klyachko. *Quantum Marginal Problem and Representations of the Symmetric Group*. 2004. arXiv: quant-ph/0409113.
- [Kly06] A. A. Klyachko. “Quantum Marginal Problem and N-representability”. In: *Journal of Physics: Conference Series* 36.1 (2006), p. 72. DOI: 10.1088/1742-6596/36/1/014.
- [KM72] V. Klee and G. J. Minty. “How Good Is the Simplex Algorithm”. In: *Inequalities* 3.3 (1972), pp. 159–175.
- [KMY04] P. Kumar, J. S. B. Mitchell, and E. A. Yildirim. “Approximate Minimum Enclosing Balls in High Dimensions Using Core-Sets”. In: *ACM Journal of Experimental Algorithmics* 8 (2004), 1.1–es. DOI: 10.1145/996546.996548.
- [KN79] G. Kempf and L. Ness. “The Length of Vectors in Representation Spaces”. In: *Algebraic Geometry*. Vol. 732. Berlin, Heidelberg: Springer Berlin Heidelberg, 1979, pp. 233–243. DOI: 10.1007/BFb0066647.
- [Knu71] D. Knutson. *Algebraic Spaces*. Vol. 203. Lecture Notes in Mathematics. Berlin, Heidelberg: Springer, 1971. DOI: 10.1007/BFb0059750.
- [Knu98] D. E. Knuth. *The Art of Computer Programming*. 2nd ed. Vol. III. Addison-Wesley, 1998. URL: <https://www.worldcat.org/oclc/312994415>.

- [KO23] R. Kothari and R. O’Donnell. “Mean Estimation When You Have the Source Code; or, Quantum Monte Carlo Methods”. In: *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics, 2023, pp. 1186–1215. doi: 10.1137/1.9781611977554.ch44.
- [Kos73] B. Kostant. “On Convexity, the Weyl Group and the Iwasawa Decomposition”. In: *Annales scientifiques de l’École Normale Supérieure*. 4th ser. 6.4 (1973), pp. 413–455. doi: 10.24033/asens.1254.
- [KP96] H. Kraft and C. Procesi. *Classical Invariant Theory, a Primer*. 1996.
- [Kru37] J. Kruithof. “Telefoonverkeersrekening”. In: *De Ingenieur* 52 (1937), E15–E25.
- [KT18] M. Karimi and L. Tunçel. “Primal-Dual Interior-Point Methods for Domain-Driven Formulations: Algorithms”. 2018. arXiv: 1804.06925.
- [KV16] E. de Klerk and F. Vallentin. “On the Turing Model Complexity of Interior Point Methods for Semidefinite Programming”. In: *SIAM Journal on Optimization* 26.3 (2016), pp. 1944–1961. doi: 10.1137/15M103114X.
- [KXY97] K. O. Kortanek, X. Xu, and Y. Ye. “An Infeasible Interior-Point Algorithm for Solving Primal and Dual Geometric Programs”. In: *Mathematical Programming* 76.1 (1997), pp. 155–181. doi: 10.1007/BF02614382.
- [Lan17] J. M. Landsberg. *Geometry and Complexity Theory*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2017. doi: 10.1017/9781108183192.
- [LCB14a] M. Lubasch, J. I. Cirac, and M.-C. Banuls. “Algorithms for Finite Projected Entangled Pair States”. In: *Physical Review B* 90.6 (2014), p. 064425.
- [LCB14b] M. Lubasch, J. I. Cirac, and M.-C. Banuls. “Unifying Projected Entangled Pair State Contractions”. In: *New Journal of Physics* 16.3 (2014), p. 033014.
- [Lee13] J. M. Lee. *Smooth Manifolds*. Springer, 2013.
- [Lee18] J. M. Lee. *Introduction to Riemannian Manifolds*. 2nd ed. 2018. Graduate Texts in Mathematics 176. Springer, 2018. doi: 10.1007/978-3-319-91755-9.
- [LG04] O. Livne and G. Golub. “Scaling by Binormalization”. In: *Numerical Algorithms* 35.1 (2004), pp. 97–120.
- [LMR+11] T. Lee, R. Mittal, B. Reichardt, R. Špalek, and M. Szegedy. “Quantum Query Complexity of State Conversion”. In: *Proceedings of 52nd IEEE Annual Symposium on Foundations of Computer Science (FOCS’11)* (2011), pp. 344–353. doi: 10.1109/FOCS.2011.75.
- [LPS15] Y. T. Lee, R. Peng, and D. A. Spielman. *Sparsified Cholesky Solvers for SDD Linear Systems*. 2015. arXiv: 1506.08204.
- [LR13] T. Lee and J. Roland. “A Strong Direct Product Theorem for Quantum Query Complexity”. In: *computational complexity* 22.2 (2013), pp. 429–462. doi: 10.1007/s00037-013-0066-8.
- [LS20] Y. T. Lee and A. Sidford. *Solving Linear Programs with Sqrt(Rank) Linear System Solves*. 2020. arXiv: 1910.08033.
- [LSW00] N. Linial, A. Samorodnitsky, and A. Wigderson. “A Deterministic Strongly Polynomial Algorithm for Matrix Scaling and Approximate Permanents”. In: *Combinatorica* 20.4 (2000), pp. 545–568. doi: 10.1007/s004930070007.
- [LY22] Z. Lai and A. Yoshise. *Riemannian Interior Point Methods for Constrained Optimization on Manifolds*. 2022. arXiv: 2203.09762.
- [Mat] Mathworks. *Balance: Diagonal Scaling to Improve Eigenvalue Accuracy*. URL: <https://www.mathworks.com/help/matlab/ref/balance.html>.
- [Mey73] K. R. Meyer. “Symmetries and Integrals in Mechanics”. In: *Dynamical Systems*. Ed. by M. M. Peixoto. Academic Press, 1973, pp. 259–272. doi: 10.1016/B978-0-12-550350-1.50025-4.
- [Mey89] W. Meyer. *Toponogov’s Theorem and Applications*. Lecture Notes, Trieste. 1989.

- [MFK94] D. Mumford, J. Fogarty, and F. Kirwan. *Geometric Invariant Theory*. 3rd ed. Vol. 34. *Ergebnisse Der Mathematik Und Ihrer Grenzgebiete (2)*. Springer-Verlag, Berlin, 1994.
- [MGP+18] A. Molnar, J. Garre-Rubio, D. Pérez-García, N. Schuch, and J. I. Cirac. “Normal Projected Entangled Pair States Generating the Same State”. In: *New Journal of Physics* 20.11 (2018), p. 113017. doi: 10.1088/1367-2630/aae9fa.
- [MGSC18] A. Molnar, Y. Ge, N. Schuch, and J. I. Cirac. “A Generalization of the Injectivity Condition for Projected Entangled Pair States”. In: *Journal of Mathematical Physics* 59.2 (2018), p. 021902.
- [MO68] A. W. Marshall and I. Olkin. “Scaling of Matrices to Achieve Specified Row and Column Sums”. In: *Numerische Mathematik* 12.1 (1968), pp. 83–90. doi: 10.1007/BF02170999.
- [MS08] K. D. Mulmuley and M. Sohoni. “Geometric Complexity Theory II: Towards Explicit Obstructions for Embeddings among Class Varieties”. In: *SIAM Journal on Computing* (2008). doi: 10.1137/080718115.
- [Mul17] K. Mulmuley. “Geometric Complexity Theory V: Efficient Algorithms for Noether Normalization”. In: *Journal of the American Mathematical Society* 30.1 (2017), pp. 225–309. doi: 10.1090/jams/864.
- [MW74] J. Marsden and A. Weinstein. “Reduction of Symplectic Manifolds with Symmetry”. In: *Reports on Mathematical Physics* 5.1 (1974), pp. 121–130. doi: 10.1016/0034-4877(74)90021-4.
- [NC02] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2002.
- [Nes18] Y. Nesterov. *Lectures on Convex Optimization*. Vol. 137. Springer, 2018. doi: 10.1007/978-3-319-91578-4.
- [Neu29] J. von Neumann. “Über die analytischen Eigenschaften von Gruppen linearer Transformationen und ihrer Darstellungen”. In: *Mathematische Zeitschrift* 30.1 (1929), pp. 3–42. doi: 10.1007/BF01187749.
- [NH15] F. Nielsen and G. Hadjeres. “Approximating Covering and Minimum Enclosing Balls in Hyperbolic Geometry”. In: *Geometric Science of Information*. Ed. by F. Nielsen and F. Barbaresco. Lecture Notes in Computer Science. Springer International Publishing, 2015, pp. 586–594. doi: 10.1007/978-3-319-25040-3_63.
- [NLD+22] J. C. Napp, R. L. La Placa, A. M. Dalzell, F. G. S. L. Brandão, and A. W. Harrow. “Efficient Classical Simulation of Random Shallow 2D Quantum Circuits”. In: *Physical Review X* 12.2 (2022), p. 021021.
- [NM84] L. Ness and D. Mumford. “A Stratification of the Null Cone Via the Moment Map”. In: *American Journal of Mathematics* 106.6 (1984), p. 1281. doi: 10.2307/2374395.
- [NN08] Y. Nesterov and A. Nemirovski. “Primal Central Paths and Riemannian Distances for Convex Sets”. In: *Foundations of Computational Mathematics* 8.5 (2008), pp. 533–560. doi: 10.1007/s10208-007-9019-4.
- [NN94] Y. Nesterov and A. Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*. Vol. 13. SIAM Studies in Applied Mathematics. SIAM, 1994. doi: 10.1137/1.9781611970791.
- [NR99] A. Nemirovski and U. Rothblum. “On Complexity of Matrix Scaling”. In: *Linear Algebra and its Applications* 302–303 (1999), pp. 435–460. doi: 10.1016/S0024-3795(99)00212-8.
- [NT02] Y. Nesterov and M. J. Todd. “On the Riemannian Geometry Defined by Self-Concordant Barriers and Interior-Point Methods”. In: *Foundations of Computational Mathematics* 2.4 (2002), pp. 333–361. doi: 10.1007/s102080010032.
- [NT05] A. Nemirovski and L. Tunçel. “Cone-Free Primal-Dual Path-Following and Potential-Reduction Polynomial Time Interior-Point Methods”. In: *Mathematical Programming* 102.2 (2005), pp. 261–294. doi: 10.1007/s10107-004-0545-4.

- [NW23] H. Nieuwboer and M. Walter. *Interior-Point Methods on Manifolds: Theory and Applications*. 2023. arXiv: 2303.04771v1.
- [NW99] A. Nayak and F. Wu. “The Quantum Query Complexity of Approximating the Median and Related Statistics”. In: *Proceedings of the 31st Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. 1999, pp. 384–393. doi: 10.1145/301250.301349.
- [OCPB16] B. O’Donoghue, E. Chu, N. Parikh, and S. Boyd. “A Primal-Dual Operator Splitting Method for Conic Optimization”. In: *Journal of Optimization Theory and Applications* 169.3 (2016), pp. 1042–1068.
- [Oga20] Y. Ogata. “A Z_2 -Index of Symmetry Protected Topological Phases with Time Reversal Symmetry for Quantum Spin Chains”. In: *Communications in Mathematical Physics* 374.2 (2020), pp. 705–734.
- [Ols16] M. Olsson. *Algebraic Spaces and Stacks*. Vol. 62. American Mathematical Society, 2016.
- [ORY17] R. Ostrovsky, Y. Rabani, and A. Yousefi. “Matrix Balancing in L_p Norms: Bounding the Convergence Rate of Osborne’s Iteration”. In: *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA’17)*. 2017, pp. 154–169. doi: 10.1137/1.9781611974782.11.
- [Osb60] E. E. Osborne. “On Pre-Conditioning of Matrices”. In: *Journal of the ACM* 7.4 (1960). doi: 10.1145/321043.321048.
- [Ost78] L. M. Ostresh. “On the Convergence of a Class of Iterative Methods for Solving the Weber Location Problem”. In: *Operations Research* 26.4 (1978), pp. 597–609. doi: 10.1287/opre.26.4.597.
- [OV08] R. Orus and G. Vidal. “Infinite Time-Evolving Block Decimation Algorithm beyond Unitary Evolution”. In: *Physical Review B* 78.15 (2008), p. 155117.
- [PBT+15] H. N. Phien, J. A. Bengua, H. D. Tuan, P. Corboz, and R. Orús. “Infinite Projected Entangled Pair States Algorithm Improved: Fast Full Update and Gauge Fixing”. In: *Physical Review B* 92.3 (2015), p. 035142.
- [PBTO12] F. Pollmann, E. Berg, A. M. Turner, and M. Oshikawa. “Symmetry Protection of Topological Phases in One-Dimensional Quantum Spin Systems”. In: *Physical review b* 85.7 (2012), p. 075125.
- [PC11] T. Pock and A. Chambolle. “Diagonal Preconditioning for First Order Primal-Dual Algorithms in Convex Optimization”. In: *Proceedings of IEEE International Conference on Computer Vision (ICCV)*. 2011, pp. 1762–1769.
- [Per23a] F. Permenter. “A Geodesic Interior-Point Method for Linear Optimization over Symmetric Cones”. In: *SIAM Journal on Optimization* 33.2 (2023), pp. 1006–1034. doi: 10.1137/20M1385019.
- [Per23b] F. Permenter. “Log-Domain Interior-Point Methods for Convex Quadratic Programming”. In: *Optimization Letters* 17.7 (2023), pp. 1613–1631. doi: 10.1007/s11590-022-01952-z.
- [PHM+22] A. Pozas-Kerstjens, S. Hernández-Santana, J. R. P. Monturiol, M. C. López, G. Scarpa, C. E. González-Guillén, and D. Pérez-García. *Physics Solutions for Machine Learning Privacy Leaks*. 2022. arXiv: 2202.12319.
- [PHOW20] T. Peng, A. W. Harrow, M. Ozols, and X. Wu. “Simulating Large Quantum Circuits on a Small Quantum Computer”. In: *Physical Review Letters* 125.15 (2020), p. 150504.
- [PMV15] H. N. Phien, I. P. McCulloch, and G. Vidal. “Fast Convergence of Imaginary Time Evolution Tensor Network Algorithms by Recycling the Environment”. In: *Physical Review B* 91.11 (2015), p. 115137. doi: 10.1103/PhysRevB.91.115137.
- [PP23] D. Pérez-García and A. Pérez-Hernández. “Locality Estimates for Complex Time Evolution in 1D”. In: *Communications in Mathematical Physics* 399.2 (2023), pp. 929–970. doi: 10.1007/s00220-022-04573-w.
- [PR69] B. N. Parlett and C. Reinsch. “Balancing a Matrix for Calculation of Eigenvalues and Eigenvectors”. In: *Numerische Mathematik* 13 (1969), pp. 293–304.

- [Pro76] C. Procesi. “The Invariant Theory of $n \times n$ Matrices”. In: *Advances in Mathematics* 19.3 (1976), pp. 306–381. doi: 10.1016/0001-8708(76)90027-X.
- [PSG+10] D. Pérez-García, M. Sanz, C. E. Gonzalez-Guillen, M. M. Wolf, and J. I. Cirac. “Characterizing Symmetries in a Projected Entangled Pair State”. In: *New Journal of Physics* 12.2 (2010), p. 025010.
- [PYHP15] F. Pastawski, B. Yoshida, D. Harlow, and J. Preskill. “Holographic Quantum Error-Correcting Codes: Toy Models for the Bulk/Boundary Correspondence”. In: *Journal of High Energy Physics* 2015.6 (2015), pp. 1–55.
- [PZ22] F. Pan and P. Zhang. “Simulation of Quantum Circuits Using the Big-Batch Tensor Network Method”. In: *Physical Review Letters* 128.3 (2022), p. 030501.
- [Raz74] J. P. Razmyslov. “Trace Identities of Full Matrix Algebras over a Field of Characteristic Zero”. In: *Mathematics of the USSR-Izvestiya* 8.4 (1974), pp. 727–760. doi: 10.1070/IM1974v008n04ABEH002126.
- [RBC21] D. Robaina, M. C. Bañuls, and J. I. Cirac. “Simulating 2+1 D Z_3 Lattice Gauge Theory with an Infinite Projected Entangled-Pair State”. In: *Physical Review Letters* 126.5 (2021), p. 050401.
- [Rei08] M. Reineke. *Moduli of Representations of Quivers*. 2008. arXiv: 0802.2147.
- [Ren01] J. Renegar. *A Mathematical View of Interior-Point Methods in Convex Optimization*. SIAM, 2001. doi: 10.1137/1.9780898718812.
- [Ren88] J. Renegar. “A Polynomial-Time Algorithm, Based on Newton’s Method, for Linear Programming”. In: *Mathematical Programming* 40.1 (1988), pp. 59–93. doi: 10.1007/BF01580724.
- [RHR+21] P. Reberntrost, Y. Hamoudi, M. Ray, X. Wang, S. Yang, and M. Santha. “Quantum Algorithms for Hedging and the Learning of Ising Models”. In: *Physical Review A* 103.1 (2021), p. 012418. doi: 10.1103/PhysRevA.103.012418.
- [Roc70] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.
- [RS89] U. G. Rothblum and H. Schneider. “Scalings of Matrices Which Have Prespecified Row Sums and Column Sums via Optimization”. In: *Linear Algebra and its Applications*. Special Issue Dedicated to Alan J. Hoffman 114–115 (1989), pp. 737–764. doi: 10.1016/0024-3795(89)90491-6.
- [RTP+20] S.-J. Ran, E. Tirrito, C. Peng, X. Chen, L. Tagliacozzo, G. Su, and M. Lewenstein. *Tensor Network Contractions: Methods and Applications to Quantum Many-Body Systems*. Vol. 964. Lecture Notes in Physics. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-34489-4.
- [Rus19] A. Rusciano. *A Riemannian Corollary of Helly’s Theorem*. 2019. arXiv: 1804.10738.
- [Sag13] B. E. Sagan. *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*. Springer Science & Business Media, 2013.
- [Sak96] T. Sakai. *Riemannian Geometry*. Vol. 149. American Mathematical Society, 1996.
- [Sat21] H. Sato. *Riemannian Optimization and Its Applications*. SpringerBriefs in Electrical and Computer Engineering. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-62391-3.
- [Sch11] U. Schollwöck. “The Density-Matrix Renormalization Group in the Age of Matrix Product States”. In: *Annals of physics* 326.1 (2011), pp. 96–192.
- [Sch20] N. Schuch. *Decidability of Periodic Tilings of the Plane*. 2020. URL: <https://mathoverflow.net/questions/121483/decidability-of-periodic-tilings-of-the-plane>.
- [Sch98] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, 1998.
- [SDC+22] Z. Shi, S. Dissanayake, P. Corboz, et al. “Discovery of Quantum Phases in the Shastry-Sutherland Compound $\text{SrCu}_2(\text{BO}_3)_2$ under Extreme Conditions of Field and Pressure”. In: *Nature Communications* 13.1 (2022), pp. 1–9.

- [SH15] S. Sra and R. Hosseini. “Conic Geometric Optimisation on the Manifold of Positive Definite Matrices”. In: *SIAM Journal on Optimization* 25.1 (2015), pp. 713–739. doi: 10.1137/140978168.
- [Shi19] Y. Shitov. “An Improved Bound for the Length of Matrix Algebras”. In: *Algebra & Number Theory* 13.6 (2019), pp. 1501–1507. doi: 10.2140/ant.2019.13.1501.
- [Sho94] P. Shor. “Algorithms for Quantum Computation: Discrete Logarithms and Factoring”. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134. doi: 10.1109/SFCS.1994.365700.
- [Sin64] R. Sinkhorn. “A Relationship between Arbitrary Positive Matrices and Doubly Stochastic Matrices”. In: *The Annals of Mathematical Statistics* 35.2 (1964), pp. 876–879.
- [Sin67] R. Sinkhorn. “Diagonal Equivalence to Matrices with Prescribed Row and Column Sums”. In: *The American Mathematical Monthly* 74.4 (1967), pp. 402–405. doi: 10.2307/2314570.
- [SK67] R. Sinkhorn and P. Knopp. “Concerning Nonnegative Matrices and Doubly Stochastic Matrices”. In: *Pacific Journal of Mathematics* 21.2 (1967), pp. 343–348. doi: 10.2140/pjm.1967.21.343.
- [SMG+20] G. Scarpa, A. Molnár, Y. Ge, J. J. García-Ripoll, N. Schuch, D. Pérez-García, and S. Iblisdir. “Projected Entangled Pair States: Fundamental Analytical and Numerical Limitations”. In: *Physical Review Letters* 125.21 (2020), p. 210504.
- [Sou67] J.-M. Souriau. “Quantification géométrique. Applications”. In: *Annales de l’I.H.P. Physique théorique* 6.4 (1967), pp. 311–341. URL: <https://eudml.org/doc/75556>.
- [SPC11] N. Schuch, D. Pérez-García, and J. I. Cirac. “Classifying Quantum Phases Using Matrix Product States and PEPS”. In: *Phys. Rev. B* 84 (2011), p. 165139.
- [SPCP13] N. Schuch, D. Poilblanc, J. I. Cirac, and D. Pérez-García. “Topological Order in the Projected Entangled-Pair States Formalism: Transfer Operator and Boundary Hamiltonians”. In: *Physical review letters* 111.9 (2013), p. 090501.
- [SPWC10] M. Sanz, D. Pérez-García, M. M. Wolf, and J. I. Cirac. “A Quantum Version of Wielandt’s Inequality”. In: *IEEE Transactions on Information Theory* 56.9 (2010), pp. 4668–4673. doi: 10.1109/TIT.2010.2054552.
- [SS15] L. Schulman and A. Sinclair. “Analysis of a Classical Matrix Preconditioning Algorithm”. In: *Proceedings of 47th Annual ACM Symposium on Theory of Computing (STOC)*. 2015, pp. 831–840.
- [SS16] E. Stoudenmire and D. J. Schwab. “Supervised Learning with Tensor Networks”. In: *Advances in Neural Information Processing Systems* 29 (2016).
- [Sto64] R. Stone. *Multiple Classifications in Social Accounting*. University of Cambridge, Department of Applied Economics, 1964.
- [Str86] V. Strassen. “The Asymptotic Spectrum of Tensors and the Exponent of Matrix Multiplication”. In: *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*. 1986, pp. 49–54. doi: 10.1109/SFCS.1986.52.
- [Str87] V. Strassen. “Relative bilinear complexity and matrix multiplication.” In: *Journal für die reine und angewandte Mathematik* 1987.375-376 (1987), pp. 406–443. doi: 10.1515/crll.1987.375-376.406.
- [Str88] V. Strassen. “The asymptotic spectrum of tensors.” In: 1988.384 (1988), pp. 102–152. doi: 10.1515/crll.1988.384.102.
- [Str91] V. Strassen. “Degeneration and complexity of bilinear maps: Some asymptotic spectra.” In: 1991.413 (1991), pp. 127–180. doi: 10.1515/crll.1991.413.127.
- [Stu08] B. Sturmfels. *Algorithms in Invariant Theory*. Springer Science & Business Media, 2008.
- [SV14] M. Singh and N. K. Vishnoi. “Entropy, Optimization and Counting”. In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 2014, pp. 50–59. doi: 10.1145/2591796.2591803.

- [SV19] D. Straszak and N. K. Vishnoi. “Maximum Entropy Distributions: Bit Complexity and Stability”. In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by A. Beygelzimer and D. Hsu. Vol. 99. PMLR. 2019, pp. 2861–2891. URL: <http://proceedings.mlr.press/v99/straszak19a.html>.
- [SW22] V. Srinivasan and A. C. Wilson. “Sufficient Conditions for Non-Asymptotic Convergence of Riemannian Optimization Methods”. In: *OPT 2022: Optimization for Machine Learning (NeurIPS 2022 Workshop)*. 2022.
- [Sze04] M. Szegedy. “Quantum Speed-up of Markov Chain Based Algorithms”. In: *45th Annual IEEE Symposium on Foundations of Computer Science*. 2004, pp. 32–41. DOI: 10.1109/FOCS.2004.53.
- [Udr94] C. Udriște. *Convex Functions and Optimization Methods on Riemannian Manifolds*. Dordrecht: Springer Netherlands, 1994. DOI: 10.1007/978-94-015-8390-9.
- [Udr97] C. Udriște. “Optimization Methods on Riemannian Manifolds”. In: *Algebras, Groups and Geometries* 14 (1997), pp. 339–359. URL: <https://cir.nii.ac.jp/crid/1573387450362206848>.
- [Val79] L. G. Valiant. “Completeness Classes in Algebra”. In: *Proceedings of the 11th Annual ACM Symposium on Theory of Computing (STOC)*. Association for Computing Machinery, 1979, pp. 249–261. DOI: 10.1145/800135.804419.
- [VC04] F. Verstraete and J. I. Cirac. *Renormalization Algorithms for Quantum-Many Body Systems in Two and Higher Dimensions*. 2004. arXiv: cond-mat/0407066.
- [VDD03] F. Verstraete, J. Dehaene, and B. De Moor. “Normal Forms and Entanglement Measures for Multipartite Quantum States”. In: *Physical Review A* 68.1 (2003), p. 012103. DOI: 10.1103/PhysRevA.68.012103.
- [VHCV16] L. Vanderstraeten, J. Haegeman, P. Corboz, and F. Verstraete. “Gradient Methods for Variational Optimization of Projected Entangled-Pair States”. In: *Physical Review B* 94.15 (2016), p. 155123.
- [Vid03] G. Vidal. “Efficient Classical Simulation of Slightly Entangled Quantum Computations”. In: *Physical Review Letters* 91.14 (2003), p. 147902. DOI: 10.1103/PhysRevLett.91.147902.
- [VPC04] F. Verstraete, D. Porras, and J. I. Cirac. “DMRG and Periodic Boundary Conditions: A Quantum Information Perspective”. In: *Phys. Rev. Lett.* 93 (2004), p. 227205.
- [VW17] M. Vergne and M. Walter. “Inequalities for Moment Cones of Finite-Dimensional Representations”. In: *Journal of Symplectic Geometry* 15.4 (2017), pp. 1209–1250. DOI: 10.4310/JSG.2017.v15.n4.a8.
- [Wal14] M. Walter. “Multipartite Quantum States and Their Marginals”. PhD thesis. ETH Zurich, 2014. DOI: 10.3929/ethz-a-010250985.
- [Wal17] N. R. Wallach. *Geometric Invariant Theory*. Universitext. Cham: Springer International Publishing, 2017. DOI: 10.1007/978-3-319-65907-7.
- [WDGC13] M. Walter, B. Doran, D. Gross, and M. Christandl. “Entanglement Polytopes: Multipartite Entanglement from Single-Particle Information”. In: *Science* 340.6137 (2013), pp. 1205–1208. DOI: 10.1126/science.1232957.
- [Wei37] E. Weiszfeld. “Sur Le Point Pour Lequel La Somme Des Distances de n Points Donnés Est Minimum”. In: *Tohoku Mathematical Journal, First Series* 43 (1937), pp. 355–386.
- [Wey46] H. Weyl. *The Classical Groups: Their Invariants and Representations*. 1. Princeton University Press, 1946.
- [Whi92] S. R. White. “Density Matrix Formulation for Quantum Renormalization Groups”. In: *Phys. Rev. Lett.* 69 (1992), p. 2863.
- [Wol22] R. de Wolf. *Quantum Computing: Lecture Notes*. 2022. arXiv: 1907.09415.
- [WS22] M. Weber and S. Sra. “Riemannian Optimization via Frank-Wolfe Methods”. In: *Mathematical Programming* (2022). DOI: 10.1007/s10107-022-01840-5.

- [XY97] G. Xue and Y. Ye. “An Efficient Algorithm for Minimizing a Sum of Euclidean Norms with Applications”. In: *SIAM Journal on Optimization* 7.4 (1997), pp. 1017–1036. doi: 10.1137/S1052623495288362.
- [Yan10] L. Yang. “Riemannian Median and Its Estimation”. In: *LMS Journal of Computation and Mathematics* 13 (2010), pp. 461–479. doi: 10.1112/S1461157020090531.
- [Yao93] A. C.-C. Yao. “Quantum Circuit Complexity”. In: *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*. 1993, pp. 352–361. doi: 10.1109/SFCS.1993.366852.
- [You91] R. M. Young. “75.9 Euler’s Constant”. In: *The Mathematical Gazette* 75.472 (1991), pp. 187–190. doi: 10.2307/3620251.
- [ZCC+17] B.-X. Zheng, C.-M. Chung, P. Corboz, et al. “Stripe Order in the Underdoped Region of the Two-Dimensional Hubbard Model”. In: *Science* 358.6367 (2017), pp. 1155–1160.
- [ZS16] H. Zhang and S. Sra. “First-Order Methods for Geodesically Convex Optimization”. In: *Conference on Learning Theory*. PMLR, 2016, pp. 1617–1638. url: <https://proceedings.mlr.press/v49/zhang16b.html>.

Abstract

This thesis is concerned with a class of computational problems known as *scaling problems*. These problems appear naturally in diverse areas of mathematics, theoretical computer science and physics. Despite a surge in interest and progress in recent years, only special cases are known to be *efficiently* solvable. In fact, the current methods are fundamentally incapable of giving efficient algorithms. The goal of this thesis is to investigate new algorithmic approaches to scaling problems, as well as expanding their domain of applicability.

In Part I, we focus on structural properties of scaling problems. These structural properties can be understood through classical results from *geometric invariant theory*, a field of study concerned with actions of algebraic groups on spaces with algebro-geometric structure. We identify a new application of this theory to tensor networks in quantum many-body physics. Projected entangled pair states (PEPS) are useful Ansätze for ground states of many-body Hamiltonians, as they can naturally be made to have the same locality structure as the Hamiltonians. The local data defining PEPS has a gauge degree of freedom, and having a canonical form for this local data is desirable from both a physical and numerical point of view. We define the *minimal canonical form* for PEPS in any dimension as the solution to a certain scaling problem, and establish its excellent theoretical properties. In particular, two tensors have a common minimal canonical form if and only if they are gauge equivalent (up to taking limits), and this is the case if and only if they define the same quantum state for any geometry. Moreover, we provide rigorous algorithms for computing the minimal canonical form, circumventing known undecidability results for PEPS on grids.

In Part II, we develop interior-point methods (IPMs) for scaling problems. The IPM framework is an extremely successful tool in modern convex optimization, providing efficient algorithms for a wide range of optimization problems. It was previously (implicitly) known that they could be used to solve a subclass of scaling problems, namely *commutative* scaling problems. However, general scaling problems are *non-commutative* and correspond to optimization problems on *symmetric spaces of non-positive curvature*, whereas the standard IPM framework only operates on domains in *flat* (Euclidean) spaces. We generalize the IPM framework to the setting where the domain has curvature, and show that this generalization is capable of capturing scaling problems, as well as other natural geometric problems. The complexity of the resulting algorithm for scaling problems matches the previous state of the art, and does not obviously suffer from the same geometric obstructions.

Lastly, in Part III, we turn towards quantum computation. This is a model of computation which relies on quantum mechanical effects, and is known to be able to surpass the power of classical computation in many (but certainly not all) contexts. We contribute to this field in the following way. We provide

improvements over the previous state of the art for two basic problems: the first is for finding all marked elements in a list, and the second is for approximating the sum of a list of non-negative numbers. Next, we turn to quantum algorithms for matrix scaling and matrix balancing. We show that several classical algorithms for these problems can be sped up quantumly, trading a lower dependence on the input size for a higher dependence on the desired precision. We also study the limitations of quantum algorithms for these problems. We prove quantum lower bounds for solving these problems with constant precision, showing that our algorithms are essentially optimal in this regime. In the high-precision regime, we also show one must query essentially the entire input, ruling out speedups over the classical state of the art.

Samenvatting

Het hoofdonderwerp van dit proefschrift is *herschalingproblemen*. Dit is een klasse van computationele problemen die natuurlijk opduiken in de wiskunde, de theoretische computerwetenschappen en de natuurkunde. Desondanks een grote hoeveelheid recente werken over dit onderwerp, weten we nog niet hoe deze problemen in het algemeen *efficiënt* opgelost kunnen worden. Het doel van dit proefschrift is om nieuwe algoritmes te ontwikkelen voor herschalingsproblemen, en het domein van toepassingen te verbreden.

In Deel I ligt de focus op structurele eigenschappen van herschalingsproblemen. Deze eigenschappen kunnen begrepen worden met behulp van klassieke resultaten uit de theorie van meetkundige invarianten, wat zich bezig houdt met acties van algebraïsche groepen op ruimtes met een algebro-geometrische structuur. Wij identificeren een nieuwe toepassing van deze theorie op tensornetwerken in de context van kwantummechanische veel-deeltjes-fysica. Geprojecteerde verstrengeld-paar toestanden (PEPS) fungeren als “goede gok” voor grondstaten van Hamiltonianen voor veel-deeltjes systemen, omdat ze op natuurlijke wijze dezelfde lokaliteitsstructuur kunnen omvatten. De lokale data van PEPS is niet uniek, en het hebben van een canonieke representatie is zowel fysisch als numeriek nuttig. Wij definiëren de *minimale canonieke vorm* voor PEPS in arbitraire fysieke dimensies, als oplossingen voor een herschalingsprobleem, en gebruiken de achterliggende theorie om de structurele eigenschappen te bestuderen. In het bijzonder hebben twee tensors een gemene minimale canonieke vorm, dan en slechts dan als ze equivalent zijn onder ijktransformaties, en dit is het geval dan en slechts dan als ze altijd dezelfde kwantumtoestand voortbrengen. Ook geven we algoritmes met rigoreuze beloftes voor het vinden van de minimale canonieke vorm, waarmee eerdere onbeslisbaarheidsresultaten omzeild kunnen worden.

In Deel II ontwikkelen we inwendige-punts methodes voor het oplossen van herschalingsproblemen. De theorie van inwendige-punts methodes is uitermate succesvol in de moderne convexe optimalisatie, en geeft efficiënte algoritmes voor een groot scala aan optimalisatieproblemen. Het was hiervoor (impliciet) bekend dat een subklasse van herschalingsproblemen, namelijk de *commutatieve*, hiermee efficiënt opgelost kan worden. Algemene herschalingsproblemen zijn daarentegen helaas niet commutatief, en corresponderen met optimalisatieproblemen op symmetrische ruimtes met niet-positieve kromming (in het commutatieve geval is er geen kromming), terwijl standaard inwendige-punts methodes alleen op ongekromde ruimtes werken. Wij generaliseren inwendige-punts methodes naar gekromde ruimtes, en laten zien dat deze generalisatie ons in staat stelt om herschalingsproblemen op te lossen met een complexiteit die de beste eerder bekende resultaten evenaart. Daarnaast zijn deze nieuwe methodes niet duidelijk ontvankelijk voor dezelfde meetkundige obstructies tot het vinden van efficiënte algoritmes, in tegenstelling tot eerder werk.

Ten laatste gaan we in Deel III in op het gebruik van kwantumcomputers voor

het oplossen van herschalingsproblemen. Dit is een computationeel model dat gebruik maakt van kwantummechanische effecten, en waarvan bekend is dat het in staat is om voorbij te gaan aan de kracht van klassieke computers in veel contexten (maar zeker niet alle). De bijdrage van dit proefschrift is hier als volgt. Ten eerste geven we verbeteringen voor twee subroutines: het vinden van alle gemarkeerde elementen in een lijst, en het schatten van een som van niet-negatieve getallen. Daarna keren we terug naar twee herschalingsproblemen, namelijk matrix herschaling en matrix balancerings; voor het oplossen hiervan gebruiken we de eerder genoemde subroutines. We laten zien dat, afhankelijk van de gewenste precisie, deze problemen op een kwantumcomputer sneller opgelost kunnen worden dan mogelijk is met een klassieke computer. Ook laten we zien dat onze kwantumalgoritmes in bepaalde regimes optimaal zijn, en dat het niet mogelijk om hoge-precisie oplossingen sneller te vinden dan mogelijk met een klassieke computer.

Acknowledgements

First and foremost, I would like to thank Michael for supervising, advising and supporting me all these years. You are one of the most supportive, thoughtful and intelligent people I have had the fortune of interacting with. Life has certainly been eventful. Nobody would have expected that 6 months after the start of my PhD, we would go from our regular in-person whiteboard discussions (of sometimes unpredictable length) to only interacting digitally. I think we managed to adapt as well as anyone could have reasonably hoped for. My work-life balance was questionable from time to time, with the scales tipping to either side, but you helped me to avoid it from getting too out of hand. Our experience with working from a distance also meant that it was much easier to adapt to your moving to Bochum, and I am very happy that this decision is working out well for you. A nice side-effect of the move is that my visits to Bochum ended up feeling like short "research holidays", where we could finally continue our distraction-free whiteboard discussions of unpredictable length. All in all, it has been a great adventure and I am really glad I decided to do a PhD with you, and I am looking forward to continuing our collaboration!

I would also like to thank Eric for hosting me at KdVI all these years. Although we did not interact often, I have come to know you as a friendly, enthusiastic and mathematically open-minded person.

I would like to express my gratitude to my doctoral committee for agreeing to read my thesis. Simultaneously, I apologize for its length; the writing process took much longer than I had hoped, and reading it must have been an order of magnitude more time-consuming. Admittedly, I should have kept it shorter, and my only defense is that it would have felt incomplete to me.

This thesis would not have existed without the efforts of my amazing co-authors. I wish to thank you all for teaching me so much (both on a scientific and non-scientific level), as well as being pleasant people to interact with. We also shared a number of experiences for which I was very glad to have your support, such as first-time arXiv "submit" button presses, and late-night Zoom meetings to finish conference submissions. I would like to thank Peter in particular for visiting us in Amsterdam for a fruitful week of discussion which led to my very first research project, for hosting Michael and me in Berlin, and for the many interesting discussions we have had about self-concordance and other topics. Joran, thanks for your boundless enthusiasm, innovativity, humor and kindness. Sander, I had a great time visiting you in Paris, and filling many chalkboards together with vague but promising ideas and calculations. Thanks also for your great sense of humor and for introducing me to some great fantasy novels. Yinan, thanks for the great rubber-ducking sessions, whether it be in person or over Zoom; I am taking the duckies with me! Freek, thanks for being an awesome, kindhearted and supportive "PhD-brother", helping me learn quantum information theory, and making some of the pictures in this thesis. Ronald, thank you for all your great

insights, your attention to detail, and your high-quality jokes; they are a great source of inspiration to me. Hiroshi, thank you for the great discussions, your openness, and suggesting to write a joint paper.

The quality of the introductory parts of the thesis was much improved by the helpful proofreading of Akshay, Jana, Jelena, Sander and Subha. I would also like to thank Ronald for pointing out a large number of typos throughout the manuscript. Any remaining typos, mistakes and puns are entirely my own responsibility.

Beyond coauthors, there are many more people who contributed indirectly to this thesis, through the many useful interactions I have had with them during meetings, longer visits and conferences over the years. Among them are Simon Apers, Matthias Christandl, Arjan Cornelissen, Levent Doğan, Fulvio Gesmundo, Cole Franks, Yassine Hamoudi, Jonas Helsen, Jonathan Leake, Vladimir Lysikov, Giulio Malavolta, Rafael Oliveira, Akshay Ramachandran, Vincent Steffan, and Jeroen Zuiddam. Rafael, thank you in particular for inviting me to Waterloo; I had a great time there, and I hope our ambitious goals will come to fruition.

My interest in research did not appear out of thin air. Dorret Boomsma and Michel Nivard, thank you for believing in me at the very earliest stage of my career, and all your guidance and support. Thomas Rot, thank you for advising me during my masters' project, a time I much enjoyed; my time working with you greatly contributed to my decision to continue doing mathematics.

The QuSoft community was one of the big contributors to my happiness during my PhD. Harry and Kareljan, I would like to thank you for all your effort in creating this wonderful place, and I wish you all the best in the future. Of course, none of this would have been possible without the support offered by Doutzen, Susanne and Victor: thank you for organizing all the stage nights, retreats, QuTeas, quizzes, dinners, et cetera. I would also like to thank Evelien, Marieke, Karin and Arzu for all their help from the KdVI side, and Janine for all the organizational help from all the way from Bochum.

The first 6 months of my time at QuSoft were most intimately shared with my officemates in the peanut butter room. I would like to thank all of you for helping me turn a 9-5 job into a 2-3 job, with all the interesting, random and sometimes totally absurd conversations we had. Our time together will live on in the bronze statue. Bonus points to all of you for not kicking me out because of all the puns. Alvaro, thank you for teaching me the basics of defense, basic quantum information theory, helping me improve my Spanish, and being so passionate about politics, movies, and life in general. Arjan, my ability to make low-quality puns was exceeded only by your ability to tolerate them. Farrokh, your level-headedness and calmness formed a nice counterbalance to the general chaos that was our office. It is a shame the peanut butter tower could not last. Subha, thank you for recommending me the Discworld series, and all the great conversations we had about life, culture and parenting. This amazing time at the office was cut short by one of those "once-in-a-lifetime" events known as COVID-19. Perhaps we shouldn't have taunted it with the corona-counter.

When working from home stopped being mandatory, I ended up sharing an office with Galina, Jelena and Yanlin. Although the individual methods were completely different, it was always a pleasure to say good morning to all of you; the time of day was no constraint. Galina, thank you for making our office greener, suggesting Zaans Huisje, and our many great conversations. Jelena, thanks for

asking me a lot of fun math questions and agreeing with my complaints about public transport in certain parts of the world. Yanlin, I am very sad that our time of being each other's rubber ducky is over; I must say it was a truly magical experience sometimes.

In the final stages of my PhD, I shared an office with Poojith and Rene. Even though I rarely appeared, I enjoyed talking and complaining to you.

Beyond the office, there were plenty of other people who made QuSoft, CWI and KdVI a lively place to be. To keep the list short, I will restrict myself to those who were young at some point during my PhD. Adam, Ailsa, Akshay, Aleksander, Alex, Alvaro, Amira, Arjan, Bas, Bjarne, Chris, Christian, Daan, Davi, David, Dmitry, Dyon, Farrokh, Filippo, Florian, Fran, Galina, Garazi, Gina, Ido, Isa, Jan, Jana, Jelena, Jonas, Joran, Jordi, Joris, Koen, Léo, Lies, Llorenç, Luca, Ludo, Lynn, Mani, Manuel, Marten, Mehrdad, Max, Max, Maxim, Niels, Nikhil, Peter, Philip, Poojith, Quinten, Randy, Rene, Ruben, Sebastian, Seenivasan, Shane, Simon, Simona, Subha, Wout, Yanlin, Yaroslav, Yfke, Yinan, you all contributed greatly to the awesome time I had here, and I will miss you all.

Part of this great experience was the regular junior meeting. Arjan and Farrokh, thank you for organizing it, and handing me the keys on your way out. The freedom to choose which cookies to get was great, if at times a bit dangerous: cookies are my weakness, and eating cookies is my strength. Lynn, thank you for bringing some sanity back to my life (in the form of mandarins). We were a great team, and I am glad we organized the junior meeting together for such a long time. Fran, thank you for taking over from me; I am sure you will do a great job.

A non-trivial portion of my time was spent on foosball, and this was often one of the highlights of the day: we would always play after lunch, sometimes before lunch, often in the late afternoon, and occasionally in the evening as well. Days at the office spent this way would be over in a flash. One starts to wonder when work was actually done, and to this day I am not entirely sure; the sport might be called a "snake in the grass" with regards to time investment. A more positive spin on this is that I would like to thank all of those who played foosball in making sure that my thesis did not get even longer. I am also deeply saddened by the fact that we lost our great friend Fernando along the way.

I would also like to thank our group members in Bochum for all the great group meetings, fun outings, dinners, and Christmas market visits we shared. Amalia, Anurudh, Massimo, Maxim, Sam, Simon, Tianwei, Vladimir, I wish you all the best, and hope to see you again soon!

Pursuing a PhD is no easy task, and I would not have made it without the help of some people in particular; whether it be through mental support, botanical garden visits, and tolerating puns and waawaawewaa's. Akshay, Alexandra, Arjan, Catherine, Garazi, Jana, Jelena, Laura, Lynn, Randy, Richard, Vjosa, thank you for being there for me during all the tough periods throughout the years. I hope you will continue to support me going forward.

Finally, I would like to thank my family, including those living elsewhere and those walking on four legs, for all their support throughout the years; I hope you are proud of this thesis (and of me). I spent a large part of my PhD working from home, and life would have been far more boring had it not been for them.