The New Linear Programming Method of Karmarkar

A. Schrijver
Department of Econometrics Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands
and
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

1. Introduction

In April 1984, at the 16th Annual ACM Symposium on Theory of Computing, Narendra Karmarkar of AT&T Bell Laboratories presented a new algorithm for linear programming. The algorithm was not only shown to be theoretically efficient (i.e., its running time is bounded by a polynomial in the input size), but was also claimed to be very fast in practice — about 50 times faster than Dantzig's classical *simplex method*, for the largest problems evaluated.

This news created much excitement among computer scientists and mathematical programmers, and subsequent reports, inter alia in *Science* magazine and on the front page of the *New York Times,* contributed to a further propagation of the sensation. Linear programming is one of the mathematical fields most applied in practice. Linear programming problems occur in such diverse areas as engineering, transportation, agriculture, distribution, scheduling, nutrition, management, and a reduction of the computer time needed would not only speed up solving linear programming problems, but also would allow one to solve larger LP-problems than before. In situations like oil processing and automatic control, quick, almost forthwith, solution of LP-problems is essential.

In 1979, L.G. Khachiyan published the first polynomial-time method for linear programming, the *ellipsoid method.* This method, though theoretically efficient, turned out to behave rather disappointingly in practice. So Karmarkar's claim that he now has found a method which is both theoretically and practically efficient, was much welcomed. Karmarkar's paper was published in the December issue of *Combinatorica* [5]. However, no computational details were given.

Next, Karmarkar was invited to give plenary lectures at two international conferences, the ORSA/TIMS-meeting in November 1984 in Dallas, and the 12th International Symposium on Mathematical Programming in August 1985 at MIT. KARMARKAR described his method and variants, explaining some of the tricks used in practice, claiming superiority of his method over the simplex method, and giving a few comparisons, but he refused to give full disclosure of test problems, computer programs and running times. This has led to much uncertainty and discussion among mathematical programmers with respect to the practical value of the new method. It led to a report 'Founding father of just a footnote?' in the *Boston Globe* of August 9, 1985:

> 'This week in Cambridge, the 28-year-old KARMARKAR came under mounting fire from his colleagues at the 12th International Symposium on Mathematical Programming. They snorted at his scientific manners, scoffed at his claims and derided his results as being everything from 'frisky' to 'majestic'. Mostly, they said that his accounts of super-fast solutions to difficult problems couldn't be replicated. ....
> ... 'He may have some wonderful method after all, but I habitually mistrust all secret mathematics', said E.M.L. BEALE, a pioneer in the commercial applications of linear programming. ...
> ... KARMARKAR himself didn't advance his cause much in a talk before an unusual plenary session of the MIT meeting of some 800 scientists from around the world. He began by observing that while mathematicians agree on what constitutes convincing proof of a mathematical proposition, there is no corresponding consensus as to what makes a persuasive presentation of experimental results — a contention that was immediately disputed by many of his listeners.'

Indeed, there is some generally accepted standard in presenting computional results. One gives (or makes available) the computer program, the test data, the type of computer, the output, and the CPU-time. In essence, the results are replicable, possibly making due allowance for the running time. Generally, one tries to give as much information as possible within the compass of a lecture or report.

Although this consensus differs from that holding in mathematics, with its strict rules for definitions, theorems and proofs, it essentially is comparable with the praxis in other branches of sciences, such as physics and chemistry, when reporting on experiments.

Karmarkar's reservedness in presenting computational details may have respectable reasons, for instance that Bell Labs has propriety of the actual computer program, which might not yet be ready as a marketable package, but the scientific community turns out to sputter if despite that a similar reservedness in making claims is not observed.

In this account of the new method I will restrict myself to the theoretical aspects.

In Section 2 and 3 we briefly discuss the simplex method and Khachiyan's ellipsoid method. In Section 4 and 5 we describe Karmarkars method, while in Sections 6 and 7 we show that the method has polynomially bounded running time.

2. Linear programming and the simplex method

The *linear programming problem* (or *LP-problem)* is as follows:

$$\text{given } A\in\mathbb{Z}^{m\times n},\ \ b\in\mathbb{Z}^{m},\ \ c\in\mathbb{Z}^{n}, \qquad (1)$$
$$\text{find a vector } x\in\mathbb{Q}^{n} \text{ attaining } \max\{c^{T}x \mid Ax\leqslant b\}.$$

So it is asked to maximize the linear function c^Tx where x ranges over the polyhedron $\{x\mid Ax\leqslant b\}$. The practical relevance of this problem was revealed in the 1940s by the work of L.V. Kantorovich, Tj.C. Koopmans and G.B. Dantzig.

In 1947 Dantzig designed his famous *simplex method* for solving (1). The idea is to make a trip over the polyhedron $P:=\{x\mid Ax\leqslant b\}$ from vertex to vertex along edges, on which c^Tx increases, until an optimum vertex is attained. The correctness of this algorithm is based on the property that if a vertex x_0 of a polytope P does *not* maximize c^Tx over P, then there exists a vertex x_1 *adjacent to* x_0 for wich $c^Tx_1>c^Tx_0$. (Here x_1 *adjacent to* x_0 means that the segment x_0x_1 forms an edge of P.)

Roots of this idea occur already in Fourier [3], describing a method for minimizing $\|Ax-b\|_\infty$ (where $\|\star\|_\infty$ denotes the maximum absolute value of the entries in $\star$):

> 'Pour atteindre promptement le point inférieur du vase, on élève en un point quelconque du plan horizontal, par exemple à l'origine des x et y, une ordonnée verticale jusqu'à la rencontre du plan le plus élevé, c'est-à-dire que parmi tous les points d'intersection que l'on trouve sur cette verticale, on choisit le plus distant du plan des x et y. Soit m_1, ce point d'intersection placé sur le plan extrême. On descend sur ce même plan depuis le point m_1 jusqu'à un point m_2 d'une arête du polyèdre, et en suivant cette arête, on descend depuis le point m_2 jusqu'au sommet m_3 commun à trois plans extrêmes. Apartir du point m_3 on continue de descendre suivant une seconde arête jusqu'à un nouveau sommet m_4, et l'on continue l'application du même procédé, en suivant toujours celle des deux arêtes qui conduit à un sommet moins élevé. On arrive ainsi très-prochainement au point le plus bas du polyèdre.'

According to Fourier, this description suffices to understand the method in more dimensions. De la Vallée Poussin [9] gave a similar method.

Dantzig [2] algebraized the method, obtaining an attractive compact scheme *(simplex tableau)* and iterative procedure *(pivoting)*, which facilitates computer implementation. This simplex method turns out to be very efficient in practice and enables one to solve LP-problems in several thousands of variables.

However, it could not be proved theoretically that the simplex method is efficient. That is, no proof has been found that the running time of the simplex method is bounded by a polynomial in the size of the problem, i.e. in

$$\sum_{i,j} \log(|a_{ij}|+1) + \sum_{i} \log(|b_i|+1) + \sum_{j} \log(|c_j|+1). \tag{2}$$

In fact, Klee and Minty [7] showed, by giving a bad class of LP-problems, that with Dantzig's pivoting rule, the simplex method can require exponential running time. Their examples have as feasible regions a deformation of the n-dimensional cube (described by $2n$ inequalities), for which Dantzig's rule leads to a trip along all 2^n vertices. Several alternative pivot selection rules have been proposed, but none of them could be proved to yield a polynomial-time method.

On the other hand, Borgwardt [1] recently gave a pivoting rule which he showed to yield a polynomial-time algorithm *on the average*, in a certain natural probabilistic model. His result very much agrees with practical experience, where data seem to be more 'random' than structural.

3. The ellipsoid method

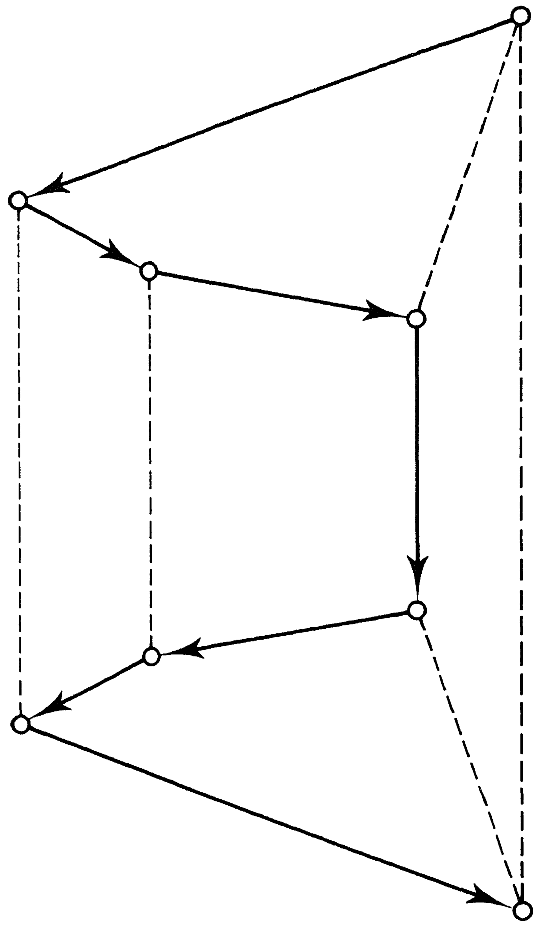
It has been an open question for a long time whether linear programming is solvable in polynomial time. Although the simplex method works well for present-day practical problems, one never knows whether the barycenter of practical problems will change, and it would then be good to have a method which can be proved to perform well always.

It was a big surprise when in 1979 the Soviet mathematician L.G. Khachiyan answered this question affirmatively, showing that the *ellipsoid method* for nonlinear programming has polynomially bounded running time when applied to LP-problems. Also this result was reported on the front page of the *New York Times.*

Khachiyan's method can be described by application to the following problem:
given

$$A \in \mathbb{Z}^{m \times n}, \quad b \in \mathbb{Z}^m, \text{ find } x \in \mathbb{Q}^n \text{ such that } Ax \leqslant b. \tag{3}$$

This problem is *polynomially equivalent* to problem (1), i.e., any polynomial-time algorithm for problem (1) yields a polynomial-time algorithm for problem (3), and conversely. Indeed, (3) easily reduces to (1) by taking $c \equiv 0$. Conversely, by the Duality theorem of linear programming, solving (1) is



A deformation of the n-dimensional cube, with a simplex path along 2^n *vertices* $(n=3)$

equivalent to solving the following system of linear inequalities:

$$Ax \leqslant b, \;\; y^T \geqslant 0, \;\; y^T A = c, \;\; y^T b \leqslant c^T x \tag{4}$$

(clearly, equations can be split into two opposite inequalities). This is a special case of (3).

To sketch Khachiyan's method, we assume that the polyhedron $\{x \mid Ax \leqslant b\}$ is bounded and full-dimensional (KHACHIYAN showed that we without loss of generality may restrict ourselves to this case). Let T be the maximum absolute value of the entries in A and b (w.l.o.g. $T \geqslant n \geqslant 2$). With Cramer's rule, one may show that the components of the vertices $\{x \mid Ax \leqslant b\}$ are at most $n^n T^n$ in absolute value. Hence $\{x \mid Ax \leqslant b\}$ is contained in the ball $E_0 := B(\mathbf{0}, R)$

around the origin of radius $R := n^{n+1}T^n$.

E_0 is the first ellipsoid. Next ellipsoids $E_1, E_2, \ldots$ are determined with the following rule. If E_k has been found, with center say z_k, check if $Az_k \leqslant b$ holds. If so, we have found a solution of $Ax \leqslant b$ as required. If not, we can choose an inequality, say $a_i^T x \leqslant b_i$ in $Ax \leqslant b$ violated by z_k. Let E_{k+1} be the ellipsoid such that

$$E_{k+1} \supseteq E_k \cap \{x \mid a_i^T x \leqslant a_i^T z_k\} \tag{5}$$

and such that E_{k+1} has smallest volume (there exist simple updating formulas for obtaining the parameters describing E_{k+1} from those describing E_k and from a_i). Since $\{x \mid Ax \leqslant b\} \subseteq \{x \mid a_i^T x \leqslant a_i^T z_k\}$, it follows by induction on k from (5) that

$$E_k \supseteq \{x \mid Ax \leqslant b\}. \tag{6}$$

Moreover, it can be proved that

$$\text{volume } E_{k+1} < e^{-1/4n} \cdot \text{volume } E_k. \tag{7}$$

Since one easily sees that volume $E_0 \leqslant (2R)^n < n^{2n^2} T^{n^2}$, inductively from (7) we have:

$$\text{volume } E_k < e^{-k/4n} \cdot n^{2n^2} T^{n^2}. \tag{8}$$

On the other hand, with Cramer's rule, using the boundedness and fulldimensionality of $\{x \mid Ax \leqslant b\}$, we know:

$$\text{volume } \{x \mid Ax \leqslant b\} \geqslant n^{-2n^2} T^{-n^2}. \tag{9}$$

(6), (8) and (9) imply:

$$n^{-2n^2} T^{-n^2} \leqslant e^{-k/4n} \cdot n^{2n^2} T^{n^2}, \tag{10}$$

i.e.,

$$k \leqslant 16n^3 \ln n \ + \ 8n^3 \ln T. \tag{11}$$

So after a polynomially bounded number of iterations we will have found a solution of $Ax \leqslant b$. Updating the ellipsoid parameters can be done in $\mathcal{O}(n^2)$ arithmetic operations, while all calculations have to be done with a precision of $\mathcal{O}(n^3 \log T)$ digits. Altogether this amounts to $\mathcal{O}(n^8 \log^2 T)$ bit operations (excluding data-handling, which takes $\mathcal{O}(\text{loglog } T \cdot \text{logloglog } T)$ for each bit operation).

Although KHACHIYAN showed the polynomial solvability of the linear programming problem, his method turned out to perform badly in practice. This is caused, among others, by the facts that the upper bound (11) of iterative steps, though polynomial in the input size, can be rather big also for moderate problems, and that the precision required to describe the successive ellipsoids is huge. (The ellipsoid method has implications in combinatorial optimization — see [4].)

Thus the question remained if there is a method for linear programming which is both practically and theoretically efficient. KARMARKAR claims that the following method is so.

4. KARMARKAR'S FORM OF THE LINEAR PROGRAMMING PROBLEM

Karmarkar's method applies to problems of the following form:

$$\begin{aligned}&\text{given } A \in \mathbb{Z}^{m\times n},\ \ c \in \mathbb{Z}^n \text{ such that } A\mathbf{1} = \mathbf{0}, \\ &\text{find } x \in \mathbb{Q}^n \text{ such that } x \geqslant 0,\ \ Ax = \mathbf{0},\ \ \mathbf{1}^T x = 1,\ \ c^T x \leqslant 0.\end{aligned} \tag{12}$$

(Here $\mathbf{1}$ denotes an all-one column vector of appropriate dimension.) This is a problem equivalent to (1) and (3). Indeed, (12) clearly is a special case of (1), as (12) amounts to finding x attaining $\max\{-c^T x \mid x \geqslant 0, Ax = \mathbf{0}, \mathbf{1}^T x = 1\}$. Conversely, (3) can be reduced to solving a system of linear equations in non-negative variables:

$$\begin{aligned}&\text{given } A \in \mathbb{Z}^{m\times n},\ \ b \in \mathbb{Z}^m, \\ &\text{find a vector } x \in \mathbb{Q}^n \text{ such that } x \geqslant 0,\ \ Ax = b.\end{aligned} \tag{13}$$

This follows by replacing $Ax \leqslant b$ by the system: $Ax' - Ax'' + x''' = b$, $x', x'', x''' \geqslant 0$. Now (13) can be reduced to (12) as follows. Let A, b as in (13) be given. Let T be the maximum absolute value of the entries in A and b. With Cramer's rule we can prove that if $x \geqslant 0$, $Ax = b$ has a solution, it has one satisfying $\mathbf{1}^T x \leqslant n^{n+1} T^n$. So we wish to solve:

$$x \geqslant 0,\ \ Ax = b,\ \ \mathbf{1}^T x \leqslant n^{n+1} T^n. \tag{14}$$

By adding one extra variable we may assume we must solve:

$$x \geqslant 0,\ \ Ax = b,\ \ \mathbf{1}^T x = n^{n+1} T^n. \tag{15}$$

By subtracting multiples of the last equation in (15) from the equations in $Ax = b$ and by scaling equations, this is equivalent to:

$$x \geqslant 0,\ \ Ax = \mathbf{0},\ \ \mathbf{1}^T x = 1. \tag{16}$$

If $A\mathbf{1} = \mathbf{0}$ then $n^{-1}\mathbf{1}$ is a solution. Otherwise, by elementary changes of the system we may assume $A\mathbf{1} = \mathbf{1}$. So we wish to find a solution x, λ for:

$$x \geqslant 0,\ \ \lambda \geqslant 0,\ \ Ax - \mathbf{1}\lambda = \mathbf{0},\ \ \mathbf{1}^T x + \lambda = 1,\ \text{ such that } \lambda \leqslant 0. \tag{17}$$

Since $A\mathbf{1} - \mathbf{1}\cdot 1 = 0$, this is a special case of (12) (taking $A := [A, -\mathbf{1}]$, $c := (0, \ldots, 0, 1)^T \in \mathbb{Z}^{n+1}$).

So Karmarkar's method applied to (12) solves linear programming in general.

5. Karmarkar's method

Karmarkar's method consists of constructing a sequence of vectors $x^0, x^1, x^2, \ldots$ converging to a solution of (12) (provided (12) has a solution). The essence of the method is to replace the condition $x \geq 0$ by a stronger condition $x \in E$, for some ellipsoid E contained in $\mathbb{R}^n_+$. As we shall see, minimizing $c^T x$ over $\{x \mid x \in E, Ax = \mathbf{0}, \mathbf{1}^T x = 1\}$ is easy, while minimizing $c^T x$ over $\{x \mid x \in \mathbb{R}^n_+, Ax = \mathbf{0}, \mathbf{1}^T x = 1\}$ is the original problem (12).

Let A and c as in (12) be given. We may assume without loss of generality that the rows of A are linearly independent, and that $n \geq 2$. Throughout we use

$$r := \sqrt{\frac{n}{n-1}}. \tag{18}$$

Let

$$x^0 := \frac{1}{n} \cdot \mathbf{1}. \tag{19}$$

So $Ax^0 = 0$, $\mathbf{1}^T x^0 = 1$, $x^0 > 0$. Next a sequence of vectors $x^0, x^1, x^2, \ldots$ such that $Ax^k = 0$, $\mathbf{1}^T x^k = 1$, $x^k > 0$ is determined, with the following recursion: denote $x^k =: (x_1^{(k)}, \ldots, x_n^{(k)})^T$, and let D be the diagonal matrix:

$$D := \operatorname{diag}(x_1^{(k)}, \ldots, x_n^{(k)}). \tag{20}$$

Define z^{k+1} and x^{k+1} as :

$$z^{k+1} \text{ is the vector attaining} \tag{21}$$

$$\min \{(c^T D)z \mid (AD)z = \mathbf{0}; \mathbf{1}^T z = n;\ z \in B(\mathbf{1}, \tfrac{1}{2}r)\},$$

$$x^{k+1} := (\mathbf{1}^T D z^{k+1})^{-1} \cdot D z^{k+1}.$$

Note that if we replace in the minimization problem the condition $z \in B(\mathbf{1}, \frac{1}{2}r)$ by the weaker condition $z \geq 0$, then we would obtain a minimization problem with optimum value at most 0 if and only if $\min \{c^T x \mid Ax = \mathbf{0},\ \mathbf{1}^T x = 1,\ x \geq 0\} \leq 0$, which is our original problem (12).

As z^{k+1} minimizes $(c^T D)z$ over the intersection of a ball with an affine space, we can write down a formula for z^{k+1}:

Proposition 1.

$$z^{k+1} = \mathbf{1} - \frac{1}{2} r \cdot \frac{(I - DA^T(AD^2A^T)^{-1}AD - n^{-1} \cdot \mathbf{1} \cdot \mathbf{1}^T)Dc}{\|(I - DA^T(AD^2A^T)^{-1}AD - n^{-1} \cdot \mathbf{1} \cdot \mathbf{1}^T)Dc\|}.$$

Proof. The minimum in (21) can be determined by projecting Dc onto the space $\{z \mid (AD)z = \mathbf{0},\ \mathbf{1}^T z = 0\}$, thus obtaining the vector:

$$p := (I - DA^T(AD^2A^T)^{-1} \mathbf{1} \cdot \mathbf{1}^T)Dc. \tag{22}$$

(Indeed, $ADp = 0$, $\mathbf{1}^T p = 0$, as one easily checks (using $AD\mathbf{1} = Ax^k = 0$), and

$c^T D - p^T$ is a linear combination of rows of AD and of the row vector $\mathbf{1}^T$.)

Then z^{k+1} is the vector reached from $\mathbf{1}$ by going over a distance $\frac{1}{2}r$ in the direction $-p$, i.e.,

$$z^{k+1} = \mathbf{1} - \frac{1}{2}r\frac{p}{\|p\|}. \quad \square \tag{23}$$

This method describes Karmarkar's method.

6. A LEMMA IN CALCULUS

In order to show correctness and convergence of the algorithm, we use the following lemma in elementary calculus. For $x = (x_1, \ldots, x_n)^T$, we denote:

$$\Pi x := x_1 \cdot \ldots \cdot x_n. \tag{24}$$

LEMMA. *Let* $n \in \mathbb{N}$, $H := \{x \in \mathbb{R}^n \mid \mathbf{1}^T x = n\}$. *Then:*

(i) $H \cap B(\mathbf{1}, r) \subseteq H \cap \mathbb{R}^n_+ \subseteq H \cap B(\mathbf{1}, (n-1)r)$;

(ii) *if* $x \in H \cap B(\mathbf{1}, \frac{1}{2}r)$, *then* $\Pi x \geqslant \frac{1}{2}(1 + \frac{\frac{1}{2}}{n-1})^{n-1}$.

(i) Let $x = (x_1, \ldots, x_n)^T \in H \cap B(\mathbf{1}, r)$. To show $x \in \mathbb{R}^n_+$, suppose without loss of generality $x_1 < 0$. Since $x \in B(\mathbf{1}, r)$ we know: $(x_2 - 1)^2 + \ldots + (x_n - 1)^2 \leqslant r^2 - (x_1 - 1)^2 < r^2 - 1 = 1/(n-1)$. Hence, with Cauchy-Schwarz:

$$(x_2 - 1) + \ldots + (x_n - 1) \leqslant \sqrt{n-1} \cdot \sqrt{(x_2 - 1)^2 + \ldots + (x_n - 1)^2} < 1.$$

Therefore, $x_1 + \ldots + x_n < x_2 + \ldots + x_n < n$, contradicting the fact that x belongs to H.

Next let $x \in H \cap \mathbb{R}^n_+$. Then

$$\begin{aligned}(x_1 - 1)^2 + \ldots + (x_n - 1)^2 &= (x_1^2 + x_n^2) - 2(x_1 + \ldots + x_n) + n \\ &\leqslant (x_1 + \ldots + x_n)^2 - 2(x_1 + \ldots + x_n) + n \\ &= n^2 - 2n + n = (n-1)^2 r^2.\end{aligned}$$

(The last inequality follows from the fact that $x \geqslant 0$.) So $x \in B(\mathbf{1}, (n-1)r)$.

(ii) We first show an auxiliary result:

$$\begin{aligned}&\text{let } \lambda, \mu \in \mathbb{R}; \text{ if } x^*, y^*, z^* \text{ achieve} \\ &\min\{xyz \mid x + y + z = \lambda,\ x^2 + y^2 + z^2 = \mu\}, \text{and} \\ &x^* \leqslant y^* \leqslant z^*, \text{ then } y^* = z^*.\end{aligned} \tag{25}$$

By replacing x, y, z by $x - \frac{1}{3}\lambda$, $y - \frac{1}{3}\lambda$, $z - \frac{1}{3}\lambda$, we may assume $\lambda = 0$.Then it is clear that the minimum is nonpositive, and hence

$x^* \leqslant 0 \leqslant y^* \leqslant z^*$. Therefore,

$$x^* y^* z^* \geqslant x^* (\frac{y^* + z^*}{2})^2 = \frac{1}{4}(x^*)^3 \geqslant -\frac{\mu}{18}\sqrt{6\mu}. \tag{26}$$

The first inequality here follows from the geometric-arithmetic mean inequality (note that $x^* \leqslant 0$), and the second inequality from the fact that if $x+y+z=0$, $x^2+y^2+z^2=\mu$, then $x \geqslant -\frac{1}{3}\sqrt{6\mu}$.
On the other hand, $(x,y,z) := (-\frac{1}{3}\sqrt{6\mu}, \frac{1}{6}\sqrt{6\mu}, \frac{1}{6}\sqrt{6\mu})$ satisfies $x+y+z=0$, $x^2+y^2+z^2=\mu$, $xyz = -\frac{\mu}{18}\sqrt{6\mu}$. Hence we have equality throughout in (2.6). Therefore, $y^* = z^*$, proving (25).

We now prove (ii) of the Lemma. The case $n=2$ being easy, assume n
Let x attain

$$\min \{\Pi x \mid x \in H \cap B(\mathbf{1}, \tfrac{1}{2}r)\}.$$

Without loss of generality, $x_1 \leqslant x_2 \leqslant \ldots \leqslant x_n$. Then for all $1 \leqslant i < j < k \leqslant n$, vector (x_i, x_j, x_k) attains

$$\min\{xyz \mid x+y+z = x_i + x_j + x_k,\ x^2+y^2+z^2 = x_i^2 + x_j^2 + x_k^2\} \tag{28}$$

(otherwise we could replace the components x_i, x_j, x_k of x by better values). Hence by (25), $x_j = x_k$. Therefore $x_2 = x_3 = \ldots = x_n$. As $x \in H \cap B(\mathbf{1}, \frac{1}{2}r)$, this implies $x_1 = \frac{1}{2}$, and $x_2 = \ldots = x_n = (1 + \frac{1}{2}/(n-1))$. This shows (ii). □

7. Operativeness of the algorithm

The operativeness of the algorithm now follows from the following proposition:

Proposition 2. *If (12) has a solution then for all* $k \geqslant 0$:

$$\frac{(c^T x^{k+1})^n}{\Pi x^{k+1}} < \frac{2}{e} \cdot \frac{(c^T x^k)^n}{\Pi x^k}. \tag{29}$$

Proof. First note:

$$\frac{(c^T x^{k+1})^n}{\Pi x^{k+1}} \cdot \frac{\Pi x^k}{(c^T x^k)^n} = \frac{(c^T D z^{k+1})^n}{\Pi (D z^{k+1})} \cdot \frac{\Pi x^k}{(c^T x^k)^n} = \left(\frac{c^T D z^{k+1}}{c^T D \mathbf{1}}\right)^n \cdot \frac{1}{\Pi z^{k+1}}. \tag{30}$$

using (21) and $\Pi(D z^{k+1}) = (\Pi x^k)(\Pi z^{k+1})$ and $x^k = D\mathbf{1}$.
We next show that, if (12) has a solution, then

$$\frac{(c^TD)z^{k+1}}{(c^TD)\mathbf{1}} \leqslant 1 - \frac{½}{n-1}. \tag{31}$$

Indeed, if (12) has a solution then $ADz=0$, $z \geqslant 0$, $c^TDz \leqslant 0$ for some $z \neq 0$. We may assume $\mathbf{1}^Tz=n$. Hence,

$$\begin{aligned} 0 &\geqslant \min\{(c^TD)z \mid z \in \mathbb{R}^n_+, ADz=\mathbf{0},\ \mathbf{1}^Tz=n\} \\ &\geqslant \min\{(c^TD)z \mid z \in B(\mathbf{1},(n-1)r), ADz=\mathbf{0}, \mathbf{1}^Tz=n\} \end{aligned} \tag{32}$$

(the last inequality follows from (i) of the Lemma).

The last minimum in (32) is attained by the vector $\mathbf{1}-(n-1)r\frac{p}{\|p\|}$, as $\mathbf{1}-\frac{1}{2}r\frac{p}{\|p\|}$ attains the minimum in (21), (cf. (22)).

Therefore, $c^TD(\mathbf{1}-(n-1)r\frac{p}{\|p\|} \leqslant 0$.

This implies

$$\begin{aligned} c^TDz^{k+1} &= c^TD(\mathbf{1}-\tfrac{1}{2}r\tfrac{p}{\|p\|}) \\ &= (1-\frac{½}{n-1})c^TD\mathbf{1} + \frac{½}{n-1}c^TD(\mathbf{1}-(n-1)r\frac{p}{\|p\|}) \\ &\leqslant (1-\frac{½}{n-1})\ (c^TD)\mathbf{1}, \end{aligned} \tag{33}$$

proving (31).

Therefore, as $\Pi z^{k+1} \geqslant \frac{1}{2}(1+\frac{1}{2}/(n-1))^{n-1}$, by (ii) of the Lemma,

$$\left[\frac{c^TDz^{k+1}}{c^TD\mathbf{1}}\right]^n \frac{1}{\Pi z^{k+1}} \leqslant (1-\frac{½}{n-1})^n \cdot \frac{1}{½(1+\frac{½}{n-1})^{n-1}} < \frac{2}{e} \tag{34}$$

(as $(1-x)/(1+x) \leqslant e^{-2x}$ for $x \geqslant 0$, since the function $e^{-2x}-(1-x)/(1+x)$ is 0 for $x=0$, and has nonnegative derivative if $x \geqslant 0$). (30) and (34) combined give (29). □

By induction on k, Proposition 2 implies, if $c^Tx^0 \geqslant 0, c^Tx^1 \geqslant 0, \ldots, c^Tx^k \geqslant 0$, (using

$$(\Pi x^k)^{1/n} \leqslant \frac{1}{n}\sum_{i=1}^{n} x_i^k = \frac{1}{n} \leqslant 1):$$

$$c^Tx^k \leqslant \frac{c^Tx^k}{(\Pi x^k)^{1/n}} < \left[\frac{2}{e}\right]^{k/n} \cdot \frac{c^Tx^0}{(\Pi x^0)^{1/n}} \leqslant \left[\frac{2}{e}\right]^{k/n} \cdot nT, \tag{35}$$

where T denotes the maximum absolute value of the entries in A and c (w.l.o.g. $T \geqslant n$).

This gives, if we take

$$N := \lceil \left(\frac{2}{1-ln2}\right) n^2 \ln(nT) \rceil, \tag{36}$$

the following theorem.

THEOREM. *If (12) has a solution, then* $c^T x^k < n^{-n} T^{-n}$ *for some* $k = 0, \ldots, N$.

PROOF. Suppose $c^T x^0, \ldots, c^T x^N \geqslant n^{-n} T^{-n}$. Then (35) holds for $k = N$, implying $c^T x^N < (2/e)^{N/n} \cdot nT \leqslant n^{-n} T^{-n}$. Contradiction. □

So suppose (12) has a solution. Then with Karmarkar's method we find a vector x^k satisfying $x^k \geqslant 0, Ax = \mathbf{0}, \mathbf{1}^T x = 1$, $c^T x^k < n^{-n} T^{-n}$. By elementary linear algebra, it is easy to find a *vertex* x^* of the polytope $\{x \geqslant 0 \mid Ax = \mathbf{0},\ \mathbf{1}^T x = 1\}$ with $c^T x^* \leqslant c^T x^k$. Hence, $c^T x^* < n^{-n} T^{-n}$. By Cramer's rule, the entries in x^* have a common denominator at most $n^n T^n$. As c is integral, this implies $c^T x^* \leqslant 0$. So x^* is a solution of (12).

Karmarkar's method consists of $\mathcal{O}(n^2 \log T)$ iterations, each consisting of $\mathcal{O}(n^3)$ arithmetic operations (due to the updating formula given by Proposition 1). All calculations have to be made with a precision of $\mathcal{O}(n^2 \log T)$ digits. Altogether this amounts to $\mathcal{O}(n^7 \log^2 T)$ bit operations (excluding data-handling, which takes $\mathcal{O}(\log\log T \cdot \log\log\log T)$ for each bit operation).

Parts of the description above are taken from the forthcoming book [8].

REFERENCES

1. K.-H. BORGWARDT (1982). The average number of pivot steps required by the simplex method is polynomial. *Zeitschrift für Operations Research 26,* 157-177.
2. G.B. DANTZIG (1951). Maximization of a linear function of variables subject to linear inequalities. TJ.C. KOOPMANS (ed.). *Activity Analysis of Production and Allocation,* John Wiley & Sons, New York, 339-347.
3. J.B.J. FOURIER (1826). Analyse des travaux de l'Académie Royale des Sciences, pendant l'année 1823, Partie mathématique. *Histoire de l'Académie Royale des Sciences de l'Institut de France 6,* xxix-xli.
4. M. GRÖTSCHEL, L. LOVÁSZ, A. SCHRIJVER (1986). *The Ellipsoid Method and Combinatorial Optimization,* Springer-Verlag, Berlin.
5. N. KARMARKAR (1984). A new polynomial-time algorithm for linear programming. *Combinatorica 4,* 373-395.
6. L.G. KHACHIYAN (1979). A polynomial algorithm in linear programming (in Russian). *Doklady Akademii Nauk SSSR 244,* 1093-1096.
7. V. KLEE, G.J. MINTY (1972). How good is the simplex algorithm? O. SHISHA (ed.). *Inequalities, III,* Academic Press, New York, 159-175.
8. A. SCHRIJVER (1986). *Theory of Linear and Integer Programming,* John

Wiley & Sons, Chichester.
9. Ch. de la Vallée Poussin (1910). Sur la méthode de l'approximation minimum. *Annales de la Société Scientifique de Bruxelles 35 (2)*, 1-16.