Spatial Redistribution of Irregularly-Spaced Pareto Fronts for More Intuitive Navigation and Solution Selection

Anton Bouter
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
Anton.Bouter@cwi.nl

Tanja Alderliesten
Academic Medical Center
Amsterdam, The Netherlands
T.Alderliesten@amc.uva.nl

Kleopatra Pirpinia
Netherlands Cancer Institute
Amsterdam, The Netherlands
K.Pirpinia@nki.nl

Peter A.N. Bosman
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
Peter.Bosman@cwi.nl

ABSTRACT

A multi-objective optimization approach is often followed by an a posteriori decision-making process, during which the most appropriate solution of the Pareto set is selected by a professional in the field. Conventional visualization methods do not correct for Pareto fronts with irregularly-spaced solutions. However, achieving a uniform spread of solutions can make the decision-making process more intuitive when decision tools such as sliders, which represent the preference for each objective, are used. We propose a method that maps an m-dimensional Pareto front to an $(m-1)$-simplex and spreads out points to achieve a more uniform distribution of these points in the simplex while maintaining the local neighborhood structure of the solutions as much as possible. This set of points can then more intuitively be navigated due to the more uniform distribution. We test our approach on a set of non-uniformly spaced 3D Pareto fronts of a real-world problem: deformable image registration of medical images. The results of these experiments are visualized as points in a triangle, showing that we indeed achieve a representation of the Pareto front with a near-uniform distribution of points where these are still positioned as expected, i.e., according to their quality in each of the objectives of interest.

CCS CONCEPTS

•**Mathematics of computing** → **Evolutionary algorithms;**

KEYWORDS

Pareto front visualization, multi-objective optimization, real-valued optimization, GOMEA

1 INTRODUCTION

Many real-world problems are actually multi-objective, meaning that multiple, often conflicting, objectives of interest arise and decisions need to be taken in the presence of trade-offs between the objectives. Without having a clear notion of preferences among these objectives a priori, the most commonly adopted approach is multi-objective optimization [8]. Solving a multi-objective optimization problem results in a set of solutions that can be considered equally good, since they represent the optimal trade-offs between the objectives. This set of solutions in the parameter space of the optimization problem is called the Pareto set, and the set of solutions in the objective space is called the Pareto front. Typically, a user then has to select a solution from the acquired Pareto front that best represents his/her preferences in terms of the objectives' trade-offs.

Naturally it follows that the way the Pareto front is presented to the user plays an essential role in enabling the final decision-making process. An appropriate Pareto front visualization should provide the user the ability to explore efficiently the entire objective space, while capturing the structure of the Pareto front and the (local) ordering between all solutions. Such a visualization can be straightforward for problems with two objectives, but presents challenges for higher dimensions. Many interesting visualization techniques have focused on visualization of Pareto fronts consisting of more than three dimensions in order to enable insightful decision making, using, e.g., level diagrams [2], self-organizing maps [7], parallel coordinates [11], the prosection method [19], or a projection to a two-dimensional (2D) [20] or three-dimensional (3D) surface [10]. Nonetheless, these methods do not address the challenge of solutions on the Pareto front being indistinguishable, which is mostly a problem for Pareto fronts that have a non-uniform distribution of solutions across their surface. An example of a uniformly spaced Pareto front (obtained from a benchmark problem), as opposed to a non-uniformly spaced Pareto front (obtained from a real-world problem), can be seen in Figure 1. Selecting a solution from the latter can be challenging, due to the fact that many solutions are virtually indistinguishable from each other. This can be a problem, because solutions that are very close in objective space could represent very different solutions in parameter space. Moreover, the
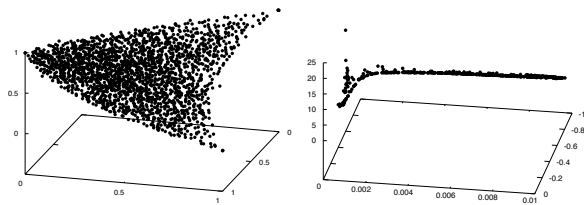
Figure 1: Example of a uniformly spaced Pareto front of a benchmark problem (left) and a non-uniformly spaced Pareto front from a real-world problem (right). Exploring the local structure of the real-world Pareto front via an interactive decision tool is potentially quite challenging.

remapping of a Pareto front for visualization purposes can even lead to a larger number of points being indistinguishable, even if such points were not close to each other in the original Pareto front. Therefore, an ideal remapping of a Pareto front to a 2D space should result in a representation where the solutions are better (i.e., more uniformly) distributed, while maintaining the characteristics of the initial topology of the Pareto front and the (local) ordering relations. This remapped set of points could then be easily navigated by using a set of trade-off sliders, one for each objective, which represent the relative preference for each objective. Since, however, there is no mapping from a higher dimensional space to a lower dimensional space that preserves 100% of local ordering relations [13], this problem becomes a multi-objective optimization problem itself. In this work, we therefore formulate this as a problem with two objectives. To solve it, we use a Multi-Objective Real-Valued Gene-pool Optimal Mixing Evolutionary Algorithm (MO-RV-GOMEA) [5]. Although the proposed approach generalizes to more than three dimensions, in this work we focus on the 3D case, for which the output of the algorithm also results in an intuitively sensible 2D visualization. Moreover, we are intrinsically motivated by data obtained from a real-world, three-objective optimization problem: Deformable Image Registration (DIR).

The remainder of this paper is organized as follows. In Section 2, we illustrate the challenges related to the uniformly distributed mapping of non-uniformly spaced sets of Pareto-optimal solutions. In Section 3, we describe our methods of addressing these challenges, including the optimization objectives and MO-RV-GOMEA. In Section 4, we describe our experimental setup, as well as our benchmark problems and the problem of DIR. In Section 4.3, we present our results, and in Section 5 we discuss our findings and present our conclusions.

2 PROBLEM DESCRIPTION

We use the example displayed in Figures 2 and 3 to illustrate the difficulties involved in achieving a sufficiently spread mapping of a Pareto front while preserving the structure of the original Pareto front as best as possible. This better illustrates the reasoning behind the use of a multi-objective approach to this problem. Figure 2 displays an example of a projection of a 3D Pareto front onto a triangle. This Pareto front contains a number of points that are largely overlapping, making the selection of one of these individual

points very difficult. Figure 3 then displays a fairly intuitive attempt at achieving a larger spread between points while preserving the structure of the original Pareto front. Because preservation of the original structure is not clearly defined, we look at the following three measures that are relevant for displaying the quality of solutions in regard to other solutions that are visualized. These measures of the mapped Pareto front should ideally be identical to those of the original Pareto front.

(1) For each point, the ordering of Euclidean distances to each other point.
(2) For each point, the set of k nearest neighbors, where k is a small constant.
(3) For each vertex of the simplex, corresponding to the extremum of one objective, the ordering of the points based on the distances from this vertex to these points.
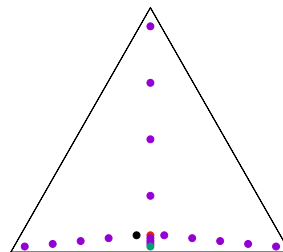

Figure 2: Projection of an arbitrary 3D Pareto front onto a triangle.
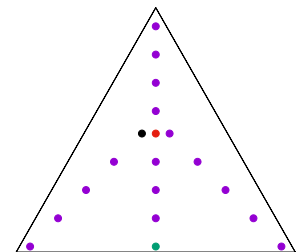

Figure 3: Example of a redistribution of the points in Figure 2.

Despite the seemingly intuitive spreading of points, all three previously listed measures are in some way different for Figure 2 and Figure 3, whereas they should ideally be similar. The first two measures have changed due to, among others, the point marked in red having two different nearest neighbors in the two figures. The third measure has changed due to, among others, the green point being further away from the bottom left vertex of the triangle than the black point in Figure 2, but closer in Figure 3.

Figures 2 and 3 only show one example, but it is likely that no sufficient spreading of the points is possible without the violation of all three measures. Moreover, it seems very difficult to define a different meaningful measure of Pareto front structure that remains constant when points are spread out. This would allow the measure to be used as a constraint while the spread between pairs of points is maximized. Instead, violations of the rankings are seemingly necessary to achieve a sufficient spread of points in the mapped Pareto front, meaning that there is an inherent trade-off between the spread of points and the preservation of the Pareto front structure. We therefore avoid using a hard constraint or penalty value for the preservation of the structure of the original Pareto front. Alternatively, a weighted sum of objectives can be optimized, but this will require the manual tuning of the weights, which is unintuitive and problem specific.

3 METHODS

An m-dimensional Pareto front $\mathcal{P}_F$ of size n, normalized to the range $[0, 1]^m$ is given as input. We assume, without loss of generality, that in the original optimization problem all objectives need to be minimized. We then construct a mapping of the points on the input Pareto front to an $(m-1)$-simplex. We consider two different representations: one where we aim to map the maximum of each dimension to the region near one of the vertices of the simplex, and one where we aim to map the minimum of each dimension to the region near one of the vertices of the simplex. Both representations are valid and could be useful in different scenarios, depending on the shape of the input Pareto front, and possible preferences regarding the decision-making process.

Points on the mapped surface (or in the mapped hypervolume for high-dimensional problems) of the simplex must be sufficiently spread in order to easily be individually selectable. At the same time, the structure of the original Pareto front must be preserved as much as possible. These two objectives are contradictive and no weights can intuitively be defined, leading us to solve this problem using a multi-objective optimization approach. The two objectives of interest are defined as a function of the set of points $\mathcal{P}_F$ of the input Pareto front. Each point $p_i \in \mathcal{P}_F$ is mapped to the point $p_i' \in [0, 1]^{m-1}$. The set of parameters of the optimization problem consists of the $m-1$ Cartesian coordinates of each p_i'. More specifically, any solution to this problem is represented by a vector $\boldsymbol{x}$ of length $\ell = n(m-1)$.

3.1 Multi-Objective Optimization

For the multi-objective approach used in this paper, the two objective functions of interest F_{spread} and F_{stress} are defined in this section, which both need to be minimized. The first objective function, F_{spread}, is aimed at the maximization of the distance between each point and its nearest neighbor. This problem is in general known as the Circles-In-A-Square (CIAS) packing problem, which is a quite difficult continuous optimization problem when solved to optimality [3]. We are however not interested in achieving optimal spread of all points, because we only aim to achieve a spread that is sufficient to spread out large clumps of points and prevents large empty spaces in the mapped representation of the input Pareto front. For this reason, we use a relaxed definition of the original CIAS objective function, which is derived in Equation 1, based on the p-norm, with $\boldsymbol{x}$ the vector of points and $d(i, j)$ the Euclidean distance between points i and j.

$$
\begin{aligned}
F_{\text{CIAS}}(\boldsymbol{x}) &= \arg\max_{\boldsymbol{x}} \left\{ \min_{i,j} d(i, j) \right\} \\
&= \arg\min_{\boldsymbol{x}} \left\{ \max_{i,j} \frac{1}{d(i, j)} \right\} \\
&= \arg\min_{\boldsymbol{x}} \left\{ \lim_{p \to \infty} \sqrt[p]{\sum_{i=0}^{|\boldsymbol{x}|-1} \sum_{j=0}^{i-1} \left| \frac{1}{d(i, j)} \right|^p} \right\} \\
&= \arg\min_{\boldsymbol{x}} \left\{ \lim_{p \to \infty} \sum_{i=0}^{|\boldsymbol{x}|-1} \sum_{j=0}^{i-1} d(i, j)^{-p} \right\}
\end{aligned}
\tag{1}
$$

The objective function that is used in our optimization approach is defined as F_{spread} in Equation 2, again with $d(a, b)$ the Euclidean distance between a and b. The sum of pairwise distances with $p = 4$ is used, because the smoothness of this objective makes it easier to optimize than an objective with a flat landscape, such as the original definition of F_{CIAS}.

$$
F_{\text{spread}} = \sum_{i=0}^{n-1} \sum_{j=0}^{i-1} d(p_i', p_j')^{-4}
\tag{2}
$$

Although any of the measures introduced in Section 2 could be used for optimization of the preservation of the original Pareto front structure, all these measures are discrete, which will increase the difficulty of their optimization. Instead we use the objective function defined as F_{stress} in Equation 3, with $d(a, b)$ again the Euclidean distance between a and b. This objective is continuous, has a smooth landscape, and is based on the Sammon stress [17]. The second term of Equation 3 has the purpose of mapping the extreme points of the input Pareto front onto the space near the corresponding vertex of the simplex. For this, we define V as a set of hyperplanes, where each $V_j \in V$ defines a hyperplane. The hyperplane V_j^{min} has the coordinate 0 in dimension j, and V_j^{max} is the hyperplane that has the coordinate 1 in dimension j. In Equation 3, V_j^{min} and V_j^{max} can be used for V_j, depending on which of the two representations discussed in Section 3 is used. The hyperplane V_j is mapped onto the point V_j', which uniquely corresponds to one of m vertices of the simplex.

$$
\begin{aligned}
F_{\text{stress}} = {} & \frac{2}{n-1} \sum_{i=0}^{n-1} \sum_{j=0}^{i-1} \left(d(p_i, p_j) - d(p_i', p_j') \right)^2 \\
& + \frac{1}{m} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \left(d(p_i, V_j) - d(p_i', V_j') \right)^2
\end{aligned}
\tag{3}
$$

To ensure that each point is mapped to a point inside the simplex, a constraint value is used that is equal to the number of points that is not inside the simplex. The constraint domination [9] technique is used for constraint handling. This means that a solution is considered to dominate any solution that has a higher constraint value.

3.2 MO-RV-GOMEA

For the optimization of the parameters that define a mapping, we use the recently introduced MO-RV-GOMEA [5], because Evolutionary Algorithms (EAs) are known to be among the state of the art for multi-objective optimization [8]. MO-RV-GOMEA has been shown to perform better than well-known state-of-the-art EAs [5], such as NSGA-II [9], especially when partial evaluations are possible, which is the case here.

MO-RV-GOMEA is a model-based EA for the optimization of real-valued variables. An adaptive elitist archive [14] is maintained to keep track of non-dominated solutions. The population is clustered, because this is known to be highly beneficial to find a good spread of solutions across the entire optimal Pareto front [4, 15]. A fraction of the solutions in the population that rank the best according to non-domination sorting [9] is selected. The variation

operator is based on a linkage model that explicitly defines subsets of problem variables, so-called linkage sets, that are considered to be dependent. A linkage model can either be learned during optimization, in which case a model is learned for each cluster separately, or it can be defined a priori, in which case it is fixed throughout the optimization process and each cluster uses the same linkage model. For each linkage set of each cluster, a multivariate normal probability distribution is estimated with maximum likelihood based on the selection. The estimated probability distribution of a linkage set is used to sample new values for the parameters included in this linkage set, which are inserted into existing solutions in the population. Only if such a modification of a subset of variables is considered to be an improvement, the modification is accepted. Otherwise, the solution is returned to its previous state. Partial modifications of solutions can be evaluated efficiently through so-called partial evaluations, which efficiently evaluate the modified objective values of a solution based on the modification of the variables. The probability distribution of a linkage set is newly estimated each generation, but it is scaled by a factor that is adapted based on improvements that are found and where they are found. We refer the interested reader to the relevant literature for further details [5].

To efficiently solve the objectives defined in Equations 2 and 3, we use a linkage model where each linkage set consists of all $(m-1)$ Cartesian coordinates of a single point $p_i \in \mathcal{P}_F$. This linkage model allows for efficient partial evaluations, because any objective value can efficiently be updated after the coordinates of one point are modified. Half of the population of MO-RV-GOMEA is initialized uniformly random in the $(m-1)$-simplex, while the other half of the population is initialized based on the coordinates of each point in the input Pareto front. Specifically, each solution in the latter half of the population is initialized such that each point $p_i \in \mathcal{P}_F$ maps to the point p_i' for which the barycentric weight of the jth vertex of the simplex is $p_i[j] \cdot U(0, 1)$, where $p_i[j]$ is the coordinate of p_i in dimension j. This latter half of the population can improve the speed of optimization due to more targeted initialization, while the former half of the population prevents premature convergence in rare cases where the targeted initialization procedure is unsuccessful at finding reasonable initial solutions.

4 EXPERIMENTS

We test the performance of our method by applying it to a set of 3D Pareto fronts. The results of these experiments can easily be visualized, because each 3D Pareto front is mapped to a 2-simplex, i.e., a triangle. Each point on the resulting mapping of a Pareto front is then color coded according to its normalized values in each of the dimensions of the original Pareto front. Specifically, each point is assigned an RGB color with the intensity of red corresponding to its coordinate in the x-dimension, the intensity of green corresponding to its coordinate in the y-dimension, and the intensity of blue corresponding to its coordinate in the z-dimension. This simultaneously allows for a visual representation of the degree by which the structure of the original Pareto front is preserved, and the degree by which points are spread out.

We perform two experiments for each benchmark problem introduced in Section 4.1, one where the maximum of each dimension is

mapped to a vertex of the triangle and one where the minimum of each dimension is mapped to a vertex of the triangle. A time limit of one hour is used for each experiment, and each experiment is performed on a desktop computer with an Intel Core i7-2600 CPU @ 3.40GHz.

4.1 Benchmark Problems

A number of irregularly-spaced Pareto fronts were previously obtained by taking a multi-objective approach to the real-world problem known as DIR, which is discussed in more detail in Section 4.2. These benchmark problems are referred to as DIR_1 through DIR_{10}. A 3D scatter plot of each DIR benchmark problem is displayed in the leftmost column of Figures 6 and 7, where each point is color coded according to the scheme introduced in Section 4. The benchmark problems consist of a number of points between approximately 400 and 1200 points, but a random subset of 100 points was used for the purpose of clearer visualization and faster optimization.

4.2 Deformable image registration

DIR [18] is a key tool in several medical processes, e.g., in radiotherapy [12]. DIR has a lot of potential since it can be used for radiotherapy planning as well as surgical planning and treatment response assessment [6]. However, DIR presents several challenges which limit its wider application in clinical practice.

Solving a DIR problem entails finding the optimal *non-linear* transformation to align two images, i.e., the optimal spatial correspondence between points in a so-called moving image and the reference image. For most general-purpose registration methods, DIR is formulated as a single-objective optimization problem, where the cost function to be optimized is a linear combination of terms that express objectives of interest. These objectives most often describe the dissimilarity between the images that needs to be minimized, but also the deformation magnitude. Although in general a certain amount of deformation is necessary in order to achieve a good match, and thereby a low value for the dissimilarity, too much deformation can result in physically incorrect deformations. Therefore, penalizing the deformation magnitude ensures avoiding such unwanted deformations. Depending on the type of registration problem, more objectives can be added; for example, for the most challenging registration problems which involve large anatomical changes, guidance information in the form of a third objective can help the registration algorithm; further, objectives which enforce local rigidity (i.e., not allowing deformation of certain regions such as bony anatomy) can be added.

However, this single-objective formulation presents a challenge, as the weights associated with the objectives that define the linear combination need to be determined beforehand, along with multiple other registration-specific parameters. This results in a time-consuming process, since the interplay between parameters, objectives of interest, and registration outcome is very complex for challenging registration problems, and the optimal configuration of parameters can be very problem-specific, leading to multiple trial-and-error attempts for each problem separately. Recently, multi-objective optimization approaches for DIR have been introduced, either by directly finding the optimal transformation that aligns the two images [1], or by optimizing the parameters (including
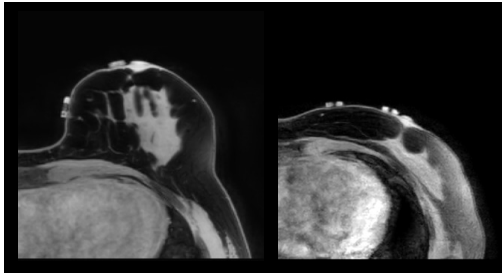
Figure 4: An example of a DIR problem. Left: slice of 3D prone MRI (moving image), right: 3D supine MRI (reference image).

the weights) of existing single-objective registration software [16]. In either case, these multi-objective approaches remove the need for predefining sets of parameters, resulting in a Pareto front of registration outcomes, which is then potentially navigable by an expert a posteriori, while immediately seeing the associated DIR outcomes.

In this work, using the aforementioned multi-objective approach that optimizes the weights associated with the objectives of an existing registration method, we acquired ten Pareto fronts of ten instances of a very challenging DIR problem: aligning breast Magnetic Resonance Imaging (MRI) scans of patients (in this study, healthy volunteers) acquired in a prone (i.e., lying face down) to a supine position (i.e., lying face up). Solving this registration problem can be useful in surgical planning for breast cancer patients that undergo breast-conserving surgery. In standard clinical practice, contrast-enhanced prone MRI is acquired to aid diagnosis, since supine MRI suffers from breathing motion artifacts which do not allow acquisition with contrast enhancement. By registering, however, the prone MRI to a non-contrast-enhanced MRI of a patient in supine position, pre-operative information is related to the intra-operative setting, where the patient is also in a supine position. This can lead to better tumor localization during surgery, reducing the chances of local recurrence and improving cosmetic outcome. However, the large deformation that the breast undergoes between prone and supine positioning makes this DIR problem very hard. For this reason, for this DIR problem not only dissimilarity and deformation magnitude were optimized, but also guidance information is exploited by minimizing the distance between marker locations in the moving and reference image that were attached on the breast of the volunteer. This resulted in ten 3D Pareto fronts. An example of a DIR problem can be seen in Figure 4.

The very different scale of these three objectives, as well as the fact that dissimilarity and guidance error are not always necessarily conflicting, results in highly non-uniform Pareto fronts (see, e.g., Figure 1 right). Another reason for an uneven distribution is possibly the difference in optimization difficulty of the three objectives. The dissimilarity objective is a highly non-convex function, with a lot of local minima, whereas the objectives that describe deformation and guidance information are convex. Each solution on the Pareto front represents a differently deformed image. Therefore, the differences between the deformed images, albeit probably small between solutions which are very close to each other, are potentially

clinically relevant depending on the DIR application, and therefore, they should be distinguishable and navigable for the user.

4.3 Experimental Results

Because we solve our problem using a multi-objective approach, the result of the optimization is itself actually a Pareto front of solutions with different trade-offs for F_{spread} and F_{stress}. In Figure 5 we show the output Pareto front for the input Pareto front of DIR_1 (shown in Figure 6), along with visualized solutions from different regions of the output Pareto front. Figures 5b and 5d are the best solutions in terms of F_{spread} and F_{stress}, respectively. The fact that these solutions are optimal in one objective also means that they are the worst solution of the Pareto front in terms of the other objective. Figures 5b and 5d make it evident that these solutions are both not desirable, i.e., Figure 5b shows a distribution of points where similarly colored points are not mapped to the same region of the triangle, and Figure 5d shows large clumps of indistinguishable points. Rather, a solution from a non-extreme region of the Pareto front should be selected, as shown in Figure 5c. Automatic selection of a solution from the Pareto front could be performed based on, e.g., a linear combination of weights of the two objectives, or a region of interest on the Pareto front, which would also mean that the problem could be solved single-objectively. Automatic selection is however difficult to tune such that it selects a proper solution for each benchmark problem, and is moreover not the focus of this work. Solutions presented in this section are therefore manually selected from the Pareto set based on visual preference.
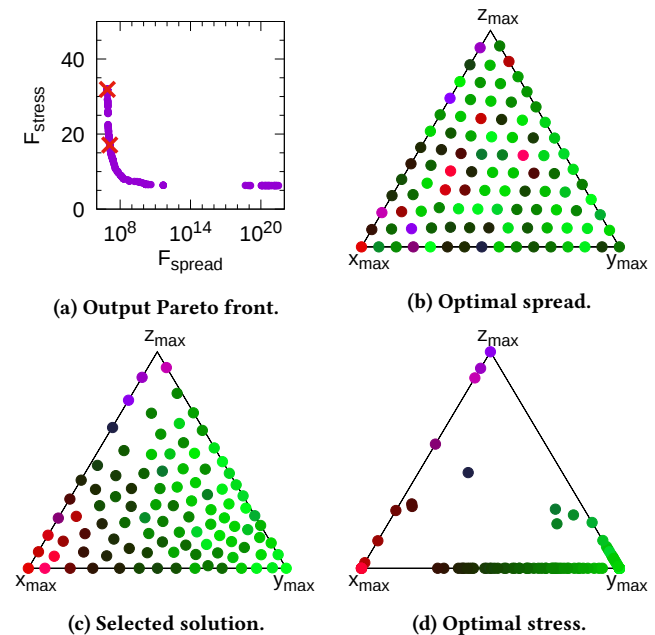

(a) Output Pareto front.

(b) Optimal spread.

(c) Selected solution.

(d) Optimal stress.

Figure 5: Output Pareto front in Figure 5a, with the marked solutions from left to right displayed in Figures 5b to 5d.

In Figures 6 and 7, we display the results of the experiments. The leftmost column shows 3D scatter plots of the input Pareto fronts. The second column from the left shows the output Pareto fronts of

both optimization approaches, i.e., where either the maximum or the minimum of of each dimension is directed towards one vertex of the simplex. In case the maximum of each dimension is directed to the vertices, $V_j^{\max}$ is used for V_j in Equation 3. In case the minimum of each dimension is directed to the vertices, $V_j^{\min}$ is used for V_j in Equation 3. The position of the selected solutions, displayed in the two rightmost columns, are denoted on the output Pareto fronts.

We see that the points in Figures 6 and 7 are all spread out near-uniformly, and the points are clearly positioned in regions near points with similar colors, i.e., similar objective values. Moreover, points are correctly placed in the region of the triangle that corresponds to their coordinates in the original Pareto front, e.g., green points were originally located near $(0, 1, 0)$ and should therefore be mapped to the region near $y_{\max}$, or a region opposing $y_{\min}$.

5 DISCUSSION AND CONCLUSIONS

We have introduced a method for the mapping of any m-dimensional Pareto front to an $(m-1)$-simplex, achieving a more uniform spread of points that still accurately represents the partial ordering of quality of the associated solutions in each of the objectives of interest. A uniform spread of points is beneficial to the decision-making process when solutions are initially assumed to be of equal preference, because this makes each point individually distinguishable. Moreover, if a decision-making tool such as a set of sliders to control the relative preference of each objective is used, a uniform spread of points will avoid navigation through very dense or sparse regions of the solution space. Sparse regions are difficult to navigate through, because many possible objective weights in such regions will correspond to the same solution. Dense regions of the solution space make it difficult to distinguish each solution while in the parameter space these solutions may well be significantly different.

A time limit of one hour was used for the experiments in this paper, but the key contribution of this paper is the general idea of solving the remapping problem multi-objectively rather than a perfectly stream-lined algorithm. A variety of methods could be applied to improve the efficiency of the optimization. For instance, a multi-resolution approach could be used that starts by performing the optimization on a small, diverse, subset of points on the original Pareto front. Remaining points are then incrementally added at a later stage of the optimization procedure, and these points can be initialized around already mapped points that are close in the original Pareto front. Secondly, more advanced methods for the initialization of the population might be possible, which would speed up the optimization by providing better initial solutions.

An alternative approach to the redistribution of Pareto fronts could entail the formulation of the problem as a discrete permutation problem. With such an approach, a Pareto front would be mapped to a set of uniformly distributed locations on the simplex, one for each point on the input Pareto front. The problem then consists of finding the best one-to-one mapping of points on the input Pareto fronts to locations on the simplex. Using this approach, a single-objective optimization method can be used, but it is not known whether this approach is capable of producing good results for any possible input Pareto front.

We have tested our approach on a set of 3D Pareto fronts from the real-world problem known as DIR. Results show that we are well able to find 2D representations of these Pareto fronts where each point is easily distinguishable and points are located at an intuitive position with respect to their quality in each of the objectives of interest, enabling the design and use of effective decision support tools, e.g., in the case of the DIR problem in this paper.

ACKNOWLEDGMENTS

REFERENCES

[1] T. Alderliesten, P.A.N. Bosman, and A. Bel. 2015. Getting the most out of additional guidance information in deformable image registration by leveraging multi-objective optimization. *Proc. SPIE* 9413 (2015), 94131R.

[2] X. Blasco, J.M. Herrero, J. Sanchis, and M. Martínez. 2008. A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization. *Information Sciences* 178, 20 (2008), 3908–3924.

[3] P.A.N. Bosman and M. Gallagher. 2016. The importance of implementation details and parameter settings in black-box optimization: a case study on Gaussian estimation-of-distribution algorithms and circles-in-a-square packing problems. *Soft Computing* (2016), 1–15.

[4] P.A.N. Bosman and D. Thierens. 2003. The balance between proximity and diversity in multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation* 7, 2 (2003), 174–188.

[5] A. Bouter, N.H. Luong, C. Witteveen, T. Alderliesten, and P.A.N. Bosman. 2017. The Multi-Objective Real-Valued Gene-Pool Optimal Mixing Evolutionary Algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2017)*. ACM.

[6] K.K. Brock, L.A. Dawson, M.B. Sharpe, D.J. Moseley, and D.A. Jaffray. 2006. Feasibility of a novel deformable image registration technique to facilitate classification, targeting, and monitoring of tumor and normal tissue. *International Journal of Radiation Oncology* Biology* Physics* 64, 4 (2006), 1245–1254.

[7] S. Chen, D. Amid, O. M. Shir, L. Limonad, D. Boaz, A. Anaby-Tavor, and T. Schreck. 2013. Self-organizing maps for multi-objective pareto frontiers. In *2013 IEEE Pacific Visualization Symposium (PacificVis)*. 153–160.

[8] K. Deb. 2001. *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons.

[9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197.

[10] A. Ibrahim, S. Rahnamayan, M. V. Martin, and K. Deb. 2016. 3D-RadVis: Visualization of Pareto front in many-objective optimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)*. 736–745.

[11] A. Inselberg and B. Dimsdale. 1990. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proceedings of the First IEEE Conference on Visualization: Visualization '90*. 361–378.

[12] M.L. Kessler. 2006. Image registration and data fusion in radiation therapy. *The British Journal of Radiology* 79, 1 (2006), S99–S108.

[13] M. Koppen and K. Yoshida. 2007. Visualization of Pareto-sets in evolutionary multi-objective optimization. In *Hybrid Intelligent Systems, 2007. HIS 2007. 7th International Conference on*. IEEE, 156–161.

[14] N.H. Luong and P.A.N. Bosman. 2012. Elitist archiving for multi-objective evolutionary algorithms: To adapt or not to adapt. In *International Conference on Parallel Problem Solving from Nature*. Springer, 72–81.

[15] M. Pelikan, K. Sastry, and D.E. Goldberg. 2005. Multiobjective hBOA, clustering, and scalability. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2005)*. ACM, 663–670.

[16] K. Pirpinia, P.A.N. Bosman, J.J. Sonke, M. van Herk, and T. Alderliesten. 2016. A first step toward uncovering the truth about weight tuning in deformable image registration. *Proc. SPIE* 9784 (2016), 978445.

[17] J.W. Sammon. 1969. A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* 100, 5 (1969), 401–409.

[18] A. Sotiras, C. Davatzikos, and N. Paragios. 2013. Deformable medical image registration: A survey. *IEEE Transactions on Medical Imaging* 32, 7 (2013), 1153–1190.

[19] T. Tušar and B. Filipič. 2015. Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method. *IEEE Transactions on Evolutionary Computation* 19, 2 (2015), 225–245.

[20] D. J. Walker, R. M. Everson, and J. E. Fieldsend. 2013. Visualising mutually non-dominating solution sets in many-objective optimisation. *IEEE Transactions on Evolutionary Computation* 17, 2 (2013), 165–184.
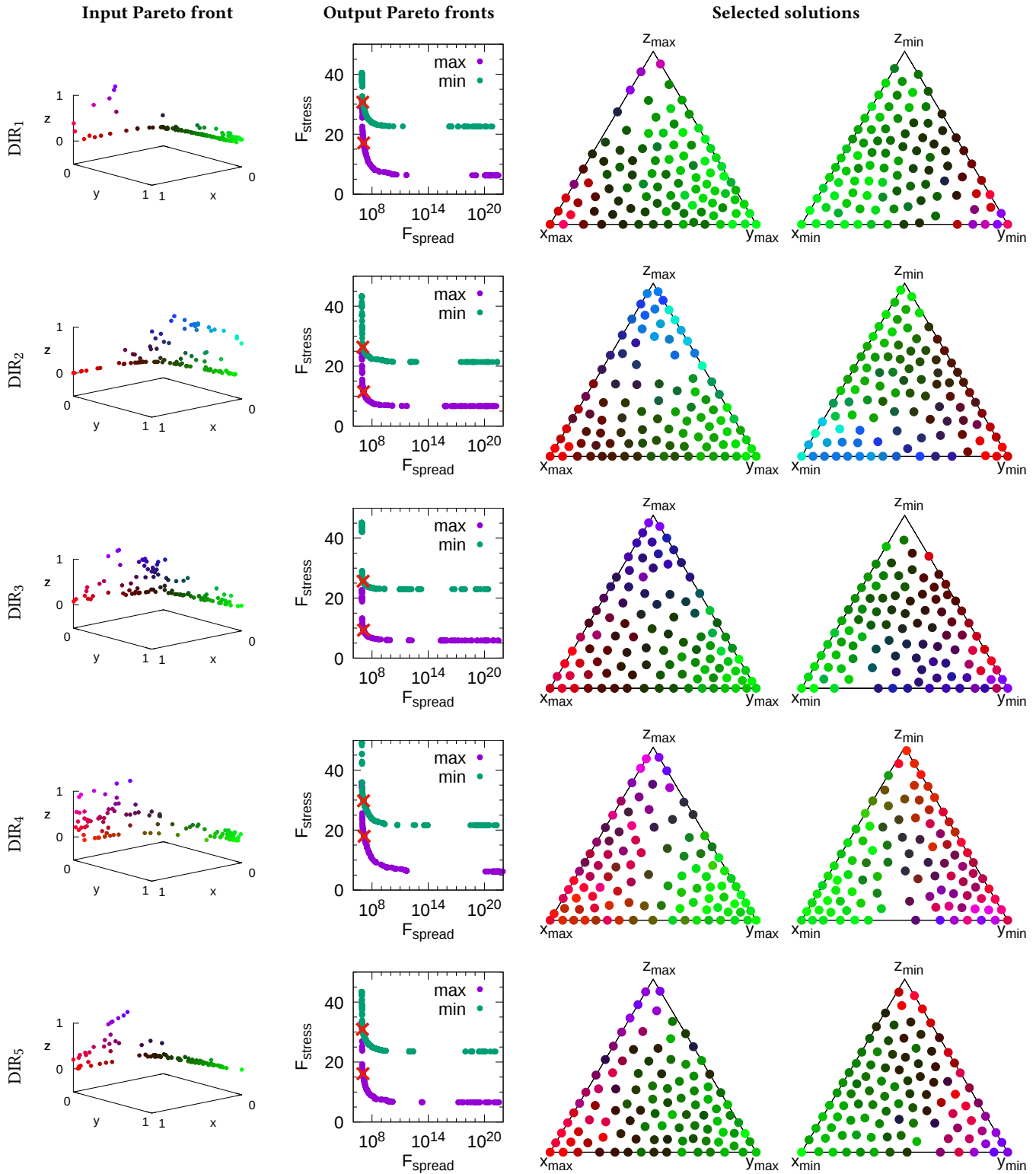
Figure 6: Results of benchmark problems DIR_1 through DIR_5. The leftmost column shows a 3D scatter plot of the input Pareto front. The second column shows output Pareto fronts of the approach with maximum values mapped to the vertices of the triangle (in purple) and the approach with minimum values mapped to the vertices of the triangle (in green). In this plot, the solutions displayed in the two rightmost columns are marked by a red cross.
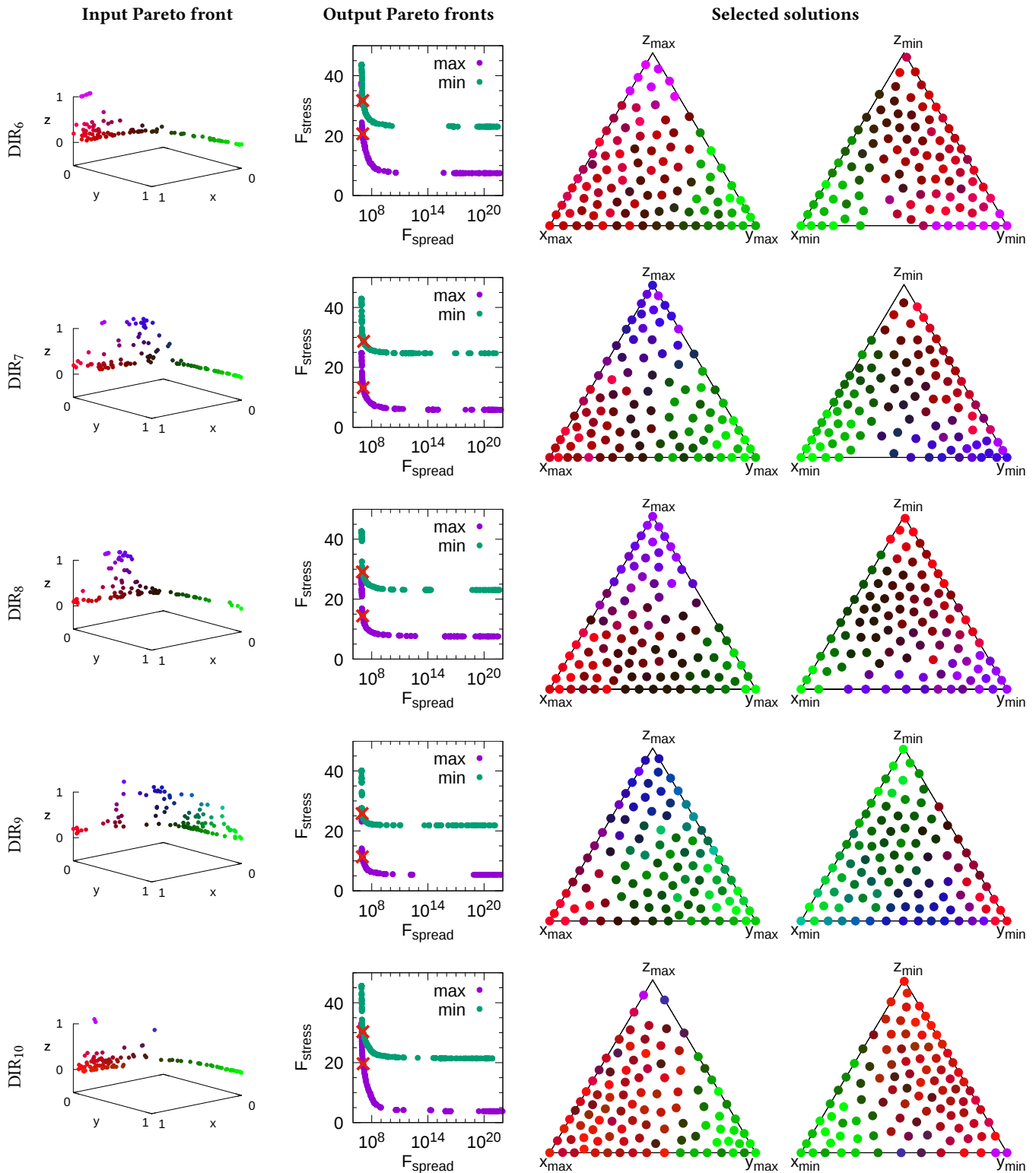
Figure 7: Results of benchmark problems DIR_6 through DIR_{10}. The leftmost column shows a 3D scatter plot of the input Pareto front. The second column shows output Pareto fronts of the approach with maximum values mapped to the vertices of the triangle (in purple) and the approach with minimum values mapped to the vertices of the triangle (in green). In this plot, the solutions displayed in the two rightmost columns are marked by a red cross.