

An Approach to Visualizing the 3D Empirical Attainment Function

Tea Tušar

Department of Intelligent Systems
Jožef Stefan Institute, and
Jožef Stefan International Postgraduate School
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
tea.tusar@ijs.si

Bogdan Filipič

Department of Intelligent Systems
Jožef Stefan Institute, and
Jožef Stefan International Postgraduate School
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
bogdan.filipic@ijs.si

ABSTRACT

When analyzing the performance of a bi-objective optimization algorithm, the empirical attainment function (EAF) is often used to visualize the attained parts of the objective space. Similarly, when comparing two algorithms, the differences in EAF values can be used to show the parts of the objective space in which the first algorithm outperforms the second one, and vice versa. This paper proposes to visualize the EAF values and differences also when assessing algorithms that optimize three criteria. This can be achieved by cutting through the 3D EAFs using multiple cutting planes and presenting the resulting intersections in 2D. The approach is described in detail and demonstrated on two artificial Pareto front approximations.

Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization

General Terms

Performance

Keywords

Visualization, Multiobjective optimization, Empirical attainment function, Attainment surface

1. INTRODUCTION

The most common means of tackling multiobjective optimization problems is using stochastic population-based algorithms, such as evolutionary algorithms [1]. A single run of such an algorithm results in a set of solutions whose images approximate the Pareto-optimal front: this is called an *approximation set*. Their stochastic nature implies that multiple runs of these algorithms result in different approximation sets. Although many measures exist to assess the quality

of approximation sets (from which the quality of the algorithm is usually inferred), visualizing them can provide additional important information. For example, visualization can be used to [6]: estimate the location, range and shape of the Pareto front, assess conflicts and trade-offs between objectives, select preferred solutions, monitor the progress or convergence of an optimization run, assess the relative performance of different optimization algorithms, etc.

While simple scatter plots can be used to visualize 2D and 3D approximation sets, visualization of multiple approximation sets is far more demanding. The *empirical attainment function* (EAF), which describes the probabilistic distribution of the approximation sets, can be used for this purpose [3]. A good example of using EAFs' values, as well as differences between EAFs for visualizing the performance of multiobjective optimization algorithms, is presented in [7]. The present paper extends the work from [7] by proposing an alternative way to visualize 2D EAFs and, more importantly, focusing on the visualization of 3D EAFs.

The paper is further organized as follows. Section 2 formally defines the EAF, followed by a brief review of the related work. Section 3 presents and demonstrates the approach for visualizing the EAF, first for the 2D case and then for the 3D one. Section 4 offers some concluding remarks.

2. BACKGROUND

2.1 Empirical Attainment Function

Consider a multiobjective optimization problem in which all criteria must be minimized. Let us first recall the *weak dominance relation* defined separately for vectors and approximation sets [5]. The objective vector \mathbf{z} weakly dominates the objective vector \mathbf{w} ($\mathbf{z} \preceq \mathbf{w}$) if and only if $z_j \leq w_j$ for all objectives j . The approximation set Z weakly dominates the approximation set W ($Z \preceq W$) if and only if every vector $\mathbf{z} \in Z$ is weakly dominated by at least one vector $\mathbf{w} \in W$.

The attainment function is based on the multiobjective concept of goal-attainment [3]: an objective vector is attained when it is weakly dominated by the approximation set returned by the multiobjective optimization algorithm. If the algorithm is run n times, each objective vector can be attained between 0 and n times. The *empirical attainment function* of the objective vector \mathbf{z} gives the frequency

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.

Copyright 2013 ACM 978-1-4503-1964-5/13/07 ...\$15.00.

of attaining \mathbf{z} by the approximation sets X_1, \dots, X_n :

$$\alpha_n(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n \mathbf{I}(X_i \preceq \{\mathbf{z}\}),$$

where \mathbf{I} is the indicator function, defined as

$$\mathbf{I}(b) = \begin{cases} 1 & \text{if } b \text{ is true,} \\ 0 & \text{otherwise,} \end{cases}$$

and \preceq is the weak dominance relation between sets.

For visualization purposes, we are often interested in areas of the objective space with a constant value of the EAF. A k %-attainment surface (also referred to as the *summary attainment surface*) consists of the tightest objective vectors that have been attained in at least k % of the runs. Such an attainment surface divides the objective space into two parts: the first is where the objective vectors have been attained in at least k % of the runs, and the second is where the objective vectors were attained in fewer than k % of the runs. When visualizing the EAF of an algorithm with n approximation sets, the *best*, *median*, and *worst* summary attainment surfaces are shown most frequently, which correspond to $\frac{1}{n}$ %, 50 % and 100 %-attainment surfaces, respectively.

In the pairwise comparison of two different algorithms, the *differences between EAFs* can be used to visualize areas in which the algorithms differ from each other the most. If only the visualization of statistically significant differences in EAF values is required, statistical tests need to be performed prior to visualization, and only the vectors with statistically significant differences need to be visualized.

2.2 Related Work

Computation of the EAF in more than two dimensions is a non-trivial task. Probably for this reason, one of the first approaches to visualizing 3D summary attainment surfaces used an even, grid-based sampling of the surface, which did not require the surface to be computed exactly [4]. More recently, an efficient algorithm for computing the exact EAF for optimization problems with three objectives was presented [2].

However, the task of visualizing an attainment surface (or equivalently, an approximation set) is quite different from visualizing EAF values and differences, since this information adds an additional dimension. Therefore, the visualization methods that are usually used for visualizing approximation sets (see [9] for a comprehensive review) cannot be used for this purpose.

To the best of our knowledge, visualization of EAF values and differences has previously been attempted only for the 2D case. In [7], for example, the difference between EAFs are plotted in order to aid the experimental analysis of algorithms on bi-objective optimization problems.

3. PROPOSED VISUALIZATION METHOD

We start by demonstrating how the 2D EAF values and differences can be visualized, and then extend this approach to the visualization in 3D.

3.1 The 2D Case

Since each summary attainment surface divides the objective space into two parts, we can visualize it either by plotting the surface or by coloring one (the dominated) part

of the space. The latter is easily done with multiple colored (or gray-shaded) rectangles that have one corner in each objective vector from the summary attainment surface and the other in the upper right corner of the visualized part of the objective space. The color (or gray shade) of the rectangles can represent the EAF value of the summary attainment surface. If this is done for all summary attainment surfaces of an algorithm in order from the best to the worst, the resulting plot provides a good idea of the distribution of the EAF values in the objective space.

In the same way, differences between EAF values for two algorithms can be visualized for each algorithm separately. Here, it is important to use colors (or gray shades) only for the positive differences for the algorithm in question (indicating the portions of the space in which this algorithm outperforms the other one), and to use white rectangles for the areas with non-positive differences.

Assume we wish to visualize the EAF values and their differences of two algorithms run n times on a given bi-objective optimization problem. The approach is as follows:

1. Combine all approximation sets and compute the overall summary attainment surfaces.
2. For each objective vector with a different EAF value, count the number of times the approximation sets of the first algorithm attained that vector, and the number of times the approximation sets of the second algorithm attained it. Mark also the differences between these two numbers.
3. Plot the computed EAF values separately for each algorithm using shaded rectangles. Plot atop the best, median, and worst summary attainment surfaces.
4. Plot the differences of the EAF values separately for each algorithm using shaded rectangles. Plot atop the best, median, and worst summary attainment surfaces.

We demonstrate the proposed visualization method on two artificial sets of approximation sets that do not result from true multiobjective optimization algorithms. This provides good control over the sets' properties, which suits our demonstration purposes. The first set of approximation sets, which is called *uniform*, contains vectors that have an almost uniform distribution in the objective space. The distribution of vectors in the second (*nonuniform*) set of approximation sets is nonuniform with only few vectors located in the middle of the approximation set and most of them near its corners. This second set contains spherical benchmark approximation sets from [9]. All approximation sets have a concave spherical shape and contain 25 vectors each, including the vectors $(0, 1)$ and $(1, 0)$, which means that the approximation sets are normalized to $[0, 1]^2$. Each set of approximation sets contains 30 approximation sets that are comparable to 30 runs of a multiobjective optimization algorithm (see Figure 1).

Figure 2 presents the best, median, and worst summary attainment surfaces for the uniform and nonuniform sets that were computed separately for each of the two sets. This figure shows how the uniform approximation sets are better at attaining the middle part of the objective space.

The EAF values and differences for these two sets are presented in Figures 3 and 4. Note that these plots show the overall summary attainment surfaces computed for both sets

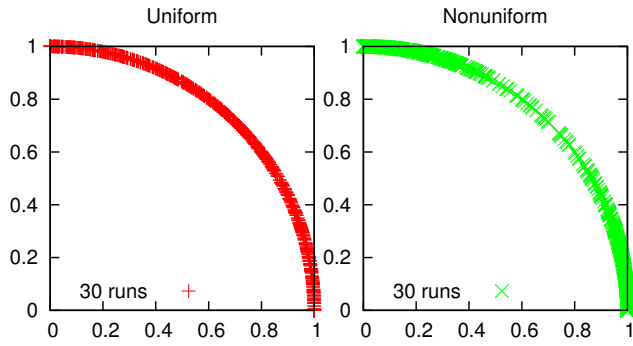


Figure 1: All vectors in the 2D uniform and nonuniform approximation sets.

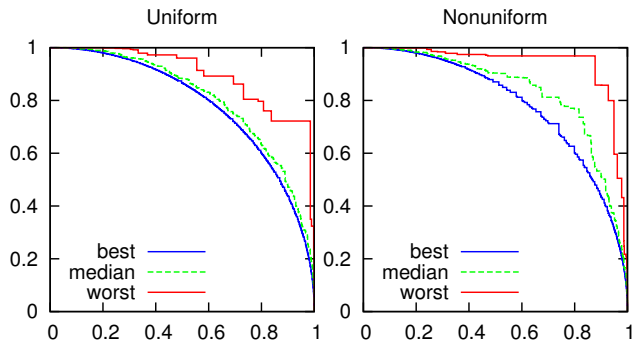


Figure 2: The best, median, and worst summary attainment surfaces generated separately for the 2D uniform and nonuniform approximation sets.

together, which is the reason why the summary attainment surfaces in these two figures are different from the ones in Figure 2. Plots of the EAF differences provide more information on the approximation sets than could be gathered from the plots in Figures 1, 2, and 3. For example, while it is again obvious that the uniform approximation sets are better at attaining the middle of the objective space, the nonuniform approximation sets receive better results at the extremities. This was not obvious from either the summary attainment surface plots or the plots of EAF values.

Our approach differs from that in [7] in that we use rectangles instead of shaded points to visualize the EAF differences. Figure 6 shows a zoomed-in region of the objective space with the EAF differences from the point of view of the uniform approximation sets, plotted first using points and then rectangles. Plotting the EAF differences using rectangles is not only the correct way to visualize them (the EAF differences are never limited to the vector in question; instead they cover an area), but it is also more clear. In fact, the points with different EAF values are not close enough to accurately represent the distribution of EAF differences in the objective space, and the presence of ‘white spots’ further hinders interpretation.

3.2 The 3D Case

The EAF values and differences can be calculated efficiently for algorithms that solve optimization problems with

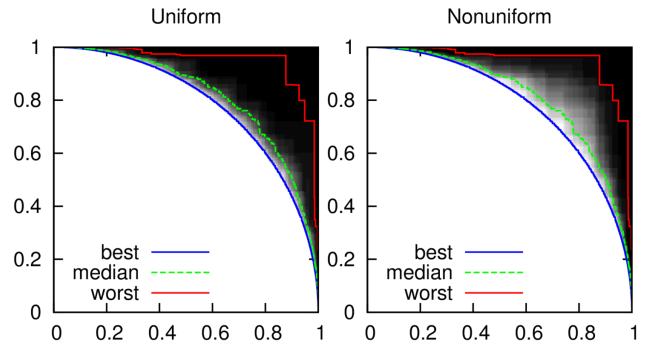


Figure 3: The EAF values for the combined 2D uniform and nonuniform approximation sets (see Figure 5 for the scale of the gray shades).

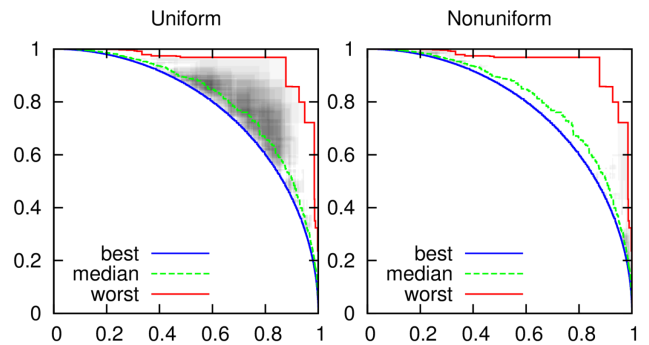


Figure 4: The differences of the EAF values between the 2D uniform and nonuniform approximation sets (see Figure 5 for the scale of the gray shades).

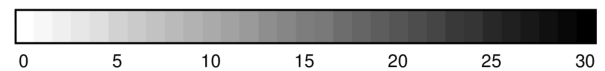


Figure 5: The scale of gray shades used in the plots throughout the paper (the absolute number between 0 and 30 means either the EAF value or the EAF difference).

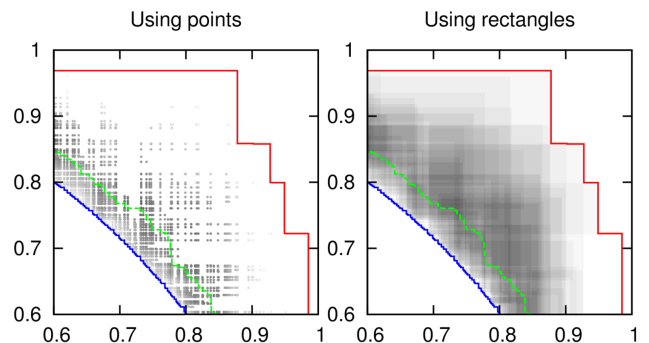


Figure 6: Comparison of the result obtained by plotting differences using points and rectangles.

three objectives, as shown in [2]. However, the visualization of such values is demanding, as they cannot simply be plotted in 3D. To solve this problem, we take inspiration from *visualization with projections* [8, 9], where cuts in the objective space are used to plot approximation sets in one fewer dimension. Essentially, the idea here is to use cutting planes to cut through the 3D objective space of the EAF values (or differences) and visualize the resulting intersections in 2D.

If we wish to visualize the EAF values and their differences of two algorithms run n times on a given three-objective optimization problem, the approach is as follows:

1. Combine all approximation sets and compute the overall summary attainment surfaces.
2. Select the cutting planes of interest.
3. Compute the intersections between each summary attainment surface and each cutting plane. Each intersection consists of the tightest objective vectors on the cutting plane, with the same EAF value as the summary attainment surface.
4. For each objective vector in the intersection, count the number of times that the approximation sets of the first algorithm attained that vector, and the number of times that the approximation sets of the second algorithm attained the vector. Mark also the differences between these two numbers.
5. Plot the computed EAF values of the vectors in the intersection using shaded rectangles for each algorithm and each cutting plane separately. Plot atop the intersections of the best, median, and worst summary attainment surfaces with the cutting plane.
6. Plot the differences of EAF values of the vectors in the intersection using shaded rectangles for each algorithm and each cutting plane separately. Plot atop the intersections of the best, median, and worst summary attainment surfaces with the cutting plane.

Figure 7 helps to explain the procedure for computing the intersections between an arbitrary attainment surface and a cutting plane. Here, the attainment surface is defined by 12 objective vectors. When the cutting plane cuts through the attainment surface, only some of the vectors (the eight that are additionally denoted as *considered vectors*) have the potential to affect the intersection. We compute the intersection between the attainment surface formed only by the considered vectors and the cutting plane and only retain the nondominated intersection vectors (the four dots from Figure 7). Figure 8 provides a more detailed step-by-step description of this procedure.

For demonstration purposes, we again use 30 *uniform* and 30 *nonuniform* concave spherical approximation sets. This time, however, each approximation set consists of 125 vectors normalized to $[0, 1]^3$. Figure 9 shows the 3D plots of the uniform and nonuniform approximation sets, together with an example of a cutting plane.

The relatively symmetrical nature of the considered approximation sets means that there is no need to visualize many cutting planes (as might be required in a general case). We chose to cut through the plane $f_1 f_2$ at three angles: 5° , 25° and 45° . We used these cutting planes and the proposed method to obtain the plots presented in Figures 10 and 11.

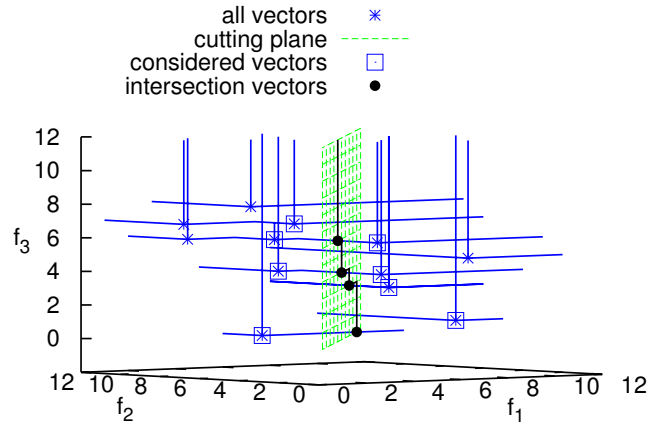


Figure 7: Intersection between the cutting plane and the attainment surface (see text for more information).

Input: The attainment surface defined by vectors $\mathbf{x}_1, \dots, \mathbf{x}_k$ and the cutting plane, which cuts the plane $f_1 f_2$ with origin $\mathbf{a} = (a_1, a_2, a_3)$ at angle φ .

1. Separate the vectors $\mathbf{x}_1, \dots, \mathbf{x}_k$ in two sets depending on whether they are located on the left or right side of the cutting plane (from the origin point of view).
2. Sort the vectors on the left of the cutting plane by their ascending value in the objective f_3 .
3. Check the set elements in this order and discard all vectors having the value in objective f_2 greater than the previous kept vector.
4. Compute the intersection between each remaining vector \mathbf{x} and the cutting plane in the following way:

$$(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \mapsto \left(a_1 + \frac{\mathbf{x}_2 - a_2}{\tan \varphi}, \mathbf{x}_2, \mathbf{x}_3 \right)$$

5. Sort the vectors on the right of the cutting plane by their ascending value in the objective f_3 .
6. Check the set elements in this order and discard all vectors having the value in objective f_1 greater than the previous kept vector.
7. Compute the intersection between each remaining vector \mathbf{x} and the cutting plane in the following way:

$$(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \mapsto (\mathbf{x}_1, a_2 + (\mathbf{x}_1 - a_1) \tan \varphi, \mathbf{x}_3)$$

8. Unite all intersection vectors in a single set and keep only the nondominated ones.
9. Perform dimension reduction by rotating the resulting intersection vectors \mathbf{x} by angle $-\varphi$:

$$(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \mapsto \left(\sqrt{(\mathbf{x}_1 - a_1)^2 - (\mathbf{x}_2 - a_2)^2}, \mathbf{x}_3 \right)$$

Output: The nondominated intersections in 2D.

Figure 8: The procedure for computing the intersection between an attainment surface and a cutting plane.

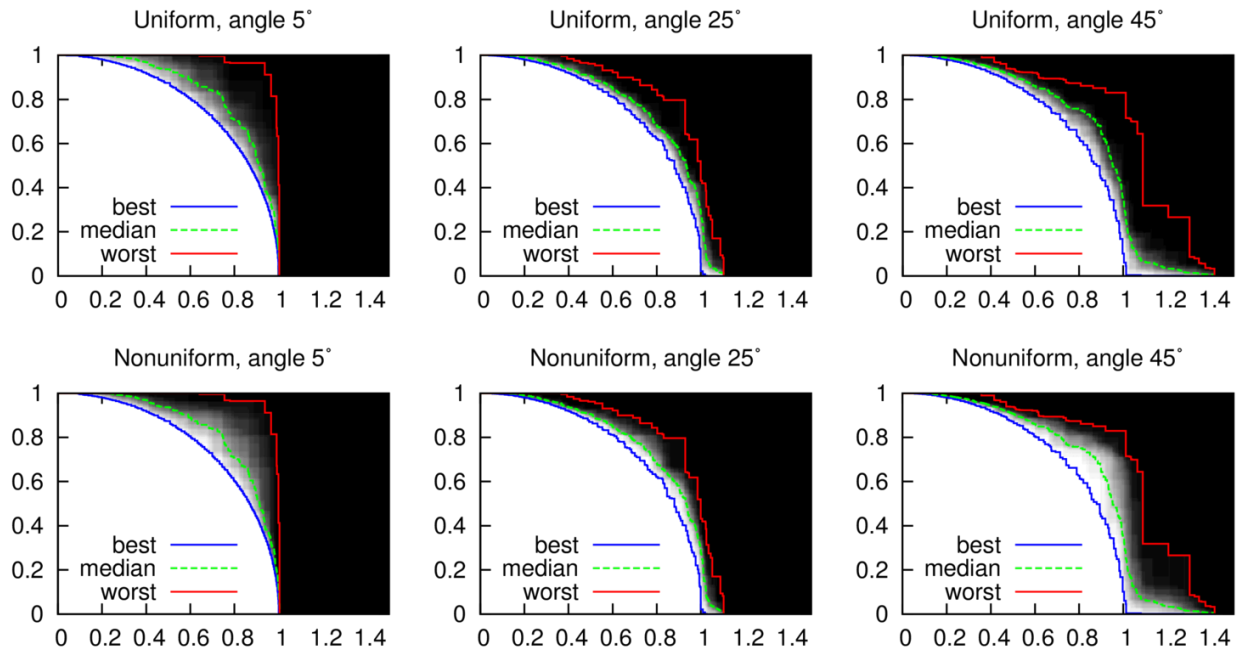


Figure 10: The EAF values for the combined 3D uniform and nonuniform approximation sets (see Figure 5 for the meaning of the gray shades).

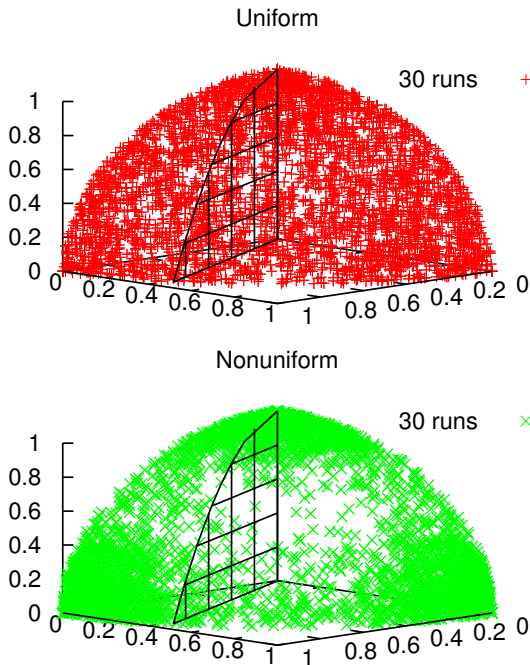


Figure 9: All vectors in the 3D uniform and nonuniform approximation sets and the cutting plane, which cuts the plane $f_1 f_2$ at angle 25° .

The plots at the 5° angle are very similar to the 2D ones. Again, the uniform approximation set is better at covering the middle of the objective space and worse at covering its extremities. At the 25° and 45° angles, the differences are

even greater in favor of the uniform approximation sets. Because the nonuniform sets are more condensed in the upper part of the objective space, we might expect them to cover this part of the objective space better than the uniform sets also at angles 25° and 45° . It is interesting to note that this is clearly not the case at angle 45° .

Finally, it is worth making a note on the complexity of the proposed method. For this purpose, we use the notation from [2], where n is the number of all approximation sets, m is the total number of objective vectors in n approximation sets, and $l = O(n^2 m)$ is the number of overall objective vectors defining the summary attainment surfaces. According to [2], it takes $O(n^2 m \log m)$ time to compute all of the summary attainment surfaces. The procedure for computing the intersections (see Figure 8) requires $O(l^2) = O(n^4 m^2)$ time in the worst case because of the nondominated sorting needed at the end. In practice, however, the required time is much smaller since the considered vectors that are included in nondominated sorting are far less than l . Therefore, in practice, the sorting of all vectors performed at the beginning of the procedure, which takes $O(l \log l)$ time in the worst case, is actually the most computationally demanding step. Counting the EAF values and differences for each algorithm would require $O(ml) = O(n^2 m^2)$ time in the worst case; however, in practice, the time spent is again much less.

Regardless of the worst case computational complexity, an average PC computed all the data needed for the presented plots within a matter of seconds. Some additional time, again measured in seconds, was needed to produce the plots¹ (plotting rectangles instead of points does take slightly longer).

¹All plots in the paper were produced using the `gnuplot` plotting program.

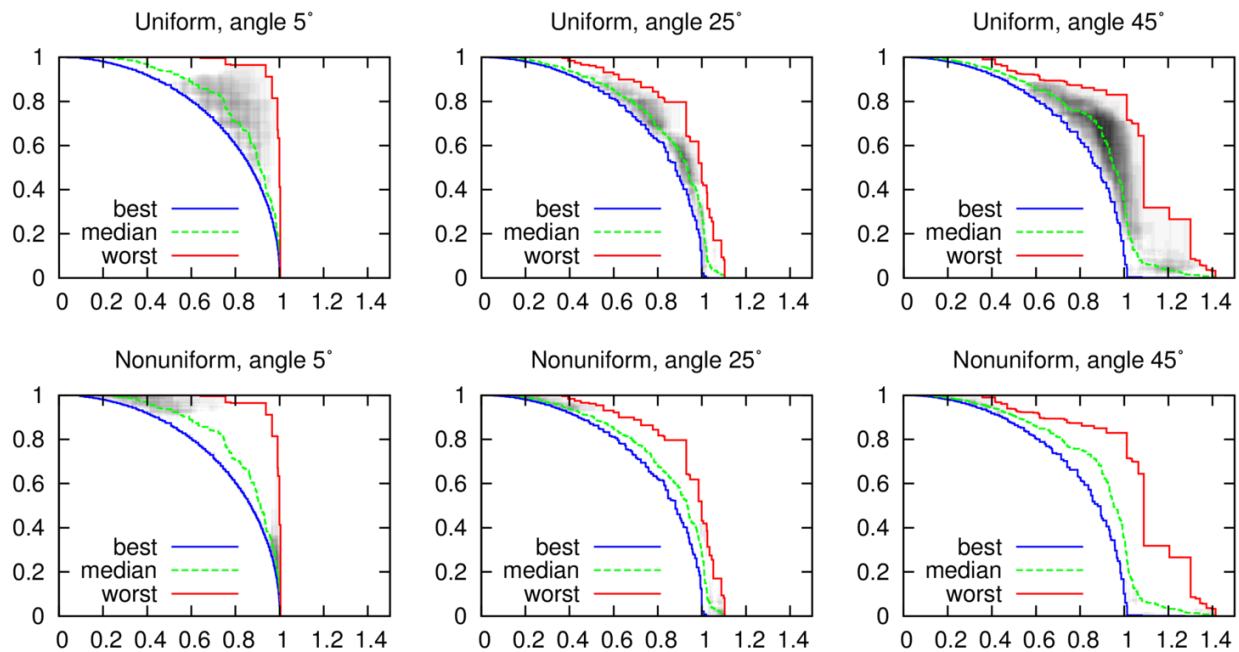


Figure 11: The differences of the EAF values between the 3D uniform and nonuniform approximation sets (see Figure 5 for the meaning of the gray shades).

4. CONCLUSION

When comparing two multiobjective optimization algorithms, it is important to be able to visualize their differences. We presented a way to do this for the 3D case by using cutting planes to cut through the 3D objective space with EAF differences and visualizing the resulting intersections in 2D. The visualization method has been described in detail and demonstrated on the comparison of uniform and nonuniform approximation sets. The complexity of this method is comparable to that of computing the summary attainment surfaces, which means it is efficient enough to be used in practice.

Our future work will use this visualization method to gain additional knowledge when comparing two algorithms on real-world optimization problems with three objectives.

5. ACKNOWLEDGMENTS

The work presented in this paper has been carried out as part of research program P2-0209 and research projects L2-3651 and J2-4120, all of which funded by the Slovenian Research Agency.

6. REFERENCES

- [1] K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, 2001.
- [2] C. M. Fonseca, A. P. Guerreiro, M. López-Ibáñez, and L. Paquete. On the computation of the empirical attainment function. In *Proceedings of the 6th International Conference on Evolutionary Multi-Criterion Optimization, EMO 2011*, pages 106–120, 2011.
- [3] V. D. Grunert da Fonseca, C. M. Fonseca, and A. O. Hall. Inferential performance assessment of stochastic optimisers and the attainment function. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization, EMO 2001*, pages 213–225, March 2001.
- [4] J. Knowles. A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers. In *Proceedings of the 5th International Conference on Intelligent Systems Design and Applications*, pages 552–557, 2005.
- [5] J. Knowles, L. Thiele, and E. Zitzler. A tutorial on the performance assessment of stochastic multiobjective optimizers. TIK-Report No. 214, Computer Engineering and Networks Laboratory, ETH Zurich, 2006.
- [6] J. Knowles, D. Corne, and K. Deb. Introduction: Problem solving, EC and EMO. In J. Knowles, D. Corne, and K. Deb, editors, *Multiobjective Problem Solving from Nature: From Concepts to Applications*, pages 1–28. Springer, 2008.
- [7] M. López-Ibáñez, L. Paquete, and T. Stützle. Exploratory analysis of stochastic local search algorithms in biobjective optimization. In T. Bartz-Beielstein, M. Chiarandini, L. Paquete, and M. Preuss, editors, *Experimental Methods for the Analysis of Optimization Algorithms*, pages 209–222. Springer-Verlag Berlin Heidelberg, 2010.
- [8] T. Tušar and B. Filipič. Visualizing 4D approximation sets of multiobjective optimizers with projections. In *Proceedings of the 13th Annual Genetic and Evolutionary Computation Conference, GECCO'11*, pages 737–744, 2011.
- [9] T. Tušar and B. Filipič. Seeing 4D Pareto front approximations. Technical Report IJS-DP 11091, Jožef Stefan Institute, 2012.