

## How to read many-objective solution sets in parallel coordinates [educational forum]

Li, Miqing; Zhen, Liangli; Yao, Xin

DOI:

[10.1109/MCI.2017.2742869](https://doi.org/10.1109/MCI.2017.2742869)

License:

None: All rights reserved

*Document Version*

Peer reviewed version

*Citation for published version (Harvard):*

Li, M, Zhen, L & Yao, X 2017, 'How to read many-objective solution sets in parallel coordinates [educational forum]', *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 88-100.  
<https://doi.org/10.1109/MCI.2017.2742869>

[Link to publication on Research at Birmingham portal](#)

### **Publisher Rights Statement:**

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

### **General rights**

Unless a licence is specified above, all rights (including copyright and moral rights) in this document are retained by the authors and/or the copyright holders. The express permission of the copyright holder must be obtained for any use of this material other than for purposes permitted by law.

- Users may freely distribute the URL that is used to identify this publication.
- Users may download and/or print one copy of the publication from the University of Birmingham research portal for the purpose of private study or non-commercial research.
- User may use extracts from the document in line with the concept of 'fair dealing' under the Copyright, Designs and Patents Act 1988 (?)
- Users may not further distribute the material nor use it for the purposes of commercial gain.

Where a licence is displayed above, please note the terms and conditions of the licence govern your use of this document.

When citing, please reference the published version.

### **Take down policy**

While the University of Birmingham exercises care and attention in making items available there are rare occasions when an item has been uploaded in error or has been deemed to be commercially or otherwise sensitive.

If you believe that this is the case for this document, please contact [UBIRA@lists.bham.ac.uk](mailto:UBIRA@lists.bham.ac.uk) providing details and we will remove access to the work immediately and investigate.

# How to Read Many-Objective Solution Sets in Parallel Coordinates

Miqing Li

School of Computer Science, University of Birmingham, Birmingham, UK

Liangli Zhen

College of Computer Science, Sichuan University, Chengdu, CHINA  
School of Computer Science, University of Birmingham, Birmingham, UK

Xin Yao

Department of Computer Science, Southern University of Science and Technology, Shenzhen, CHINA  
School of Computer Science, University of Birmingham, Birmingham, UK

**Abstract**—Rapid development of evolutionary algorithms in handling many-objective optimization problems requires viable methods of visualizing a high-dimensional solution set. The parallel coordinates plot which scales well to high-dimensional data is such a method, and has been frequently used in evolutionary many-objective optimization. However, the parallel coordinates plot is not as straightforward as the classic scatter plot to present the information contained in a solution set. In this paper, we make some observations of the parallel coordinates plot, in terms of comparing the quality of solution sets, understanding the shape and distribution of a solution set, and reflecting the relation between objectives. We hope that these observations could provide some guidelines as to the proper use of the parallel coordinates plot in evolutionary many-objective optimization.

## I. INTRODUCTION

The classic scatter plot is a basic tool in viewing solution vectors in multi-objective optimization. It allows us to observe/perceive the quality of a solution set, the shape and distribution of a solution set, the relation between objectives (e.g., the extent of their conflict), etc. Unfortunately, the scatter plot may only be drawn readily in a 2D or 3D Cartesian coordinate space. It could be difficult for people to comprehend the scatter plot in a higher-dimensional space.

An alternative to view data with four or more dimensions is using parallel coordinates [1], [2], [3] (aka value paths [4]). Parallel coordinates display multi-dimensional data (a set of vectors) in a two-dimensional graph, with each dimension of the original data being translated onto a vertical axis in the graph. A vector is represented as a polyline with vertices on the axes. As a visualization tool, parallel coordinates have received modest attention in the early stage of evolutionary multiobjective optimization (EMO) [5], [6]. As many-objective optimization (i.e., an optimization problem with more than three objectives [7], [8]) becomes a new research topic in the EMO area, there

has been increasing interest in presenting solution vectors in the high-dimensional space. Parallel coordinates which are scalable to objective dimensionality naturally become a good alternative to do so [9]. Now the parallel coordinates plot has been dominantly used in many-objective optimization despite the emergence of various visualization techniques [4], [10], [11], [12]. This includes it being used to investigate the search behavior of algorithms [13], [14], [15], to examine preference-based search [16], [17], [18], to compare different solution sets [19], [20], [21], to verify performance metrics [22], [23], [24], and furthermore to help design new many-objective optimizers [25], [26].

Despite the popularity, the parallel coordinates plot is not as straightforward as the scatter plot in presenting the information contained in a solution set. Due to mapping multi-dimensional data onto a lower 2D space, the loss of information is inevitable. This could naturally lead to several questions; specifically, in the context of multi-objective optimization, one may ask

- Can the parallel coordinates plot indicate the quality of a solution set, e.g., its convergence, uniformity and coverage?
- Can the parallel coordinates plot give insights into the shape and distribution of a solution set? In other words, what can we see from the pattern of solution lines in parallel coordinates?
- How much information can the parallel coordinates plot tell in terms of the relation among objectives? To be specific, does the order of objectives displayed in parallel coordinates matter?

In this paper, we make some observations on the above questions, attempting to provide some guidelines as to the use of parallel coordinates in evolutionary multi-objective optimization. The rest of the paper is organized as follows. Section II briefly introduces parallel coordinates. Sections III–V are devoted to answering those three questions, respectively.

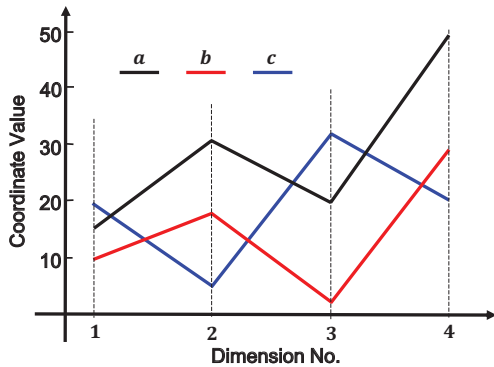


Fig. 1. The parallel coordinates plot of three four-dimensional points  $a$  (15, 31, 20, 50),  $b$  (10, 18, 2, 30) and  $c$  (20, 5, 32, 20).

Section VI describes how to draw a parallel coordinates plot. Section VII concludes the paper and presents some possible future research lines.

## II. PARALLEL COORDINATES

To show a set of points in an  $m$ -dimensional space, parallel coordinates map them onto a 2D graph, with  $m$  parallel axes being plotted, typically vertical and equally spaced. A point in an  $m$ -dimensional space is represented as a polyline with vertices on these parallel axes, and the position of the vertex on the  $i$ -th axis corresponds to the value of the point on the  $i$ -th dimension. Parallel coordinates are simple to construct and scale well with the dimensionality of data. Adding more dimensions only involves adding more axes. Figure 1 presents an example of the parallel coordinates plot, where three 4D points are mapped to three polylines, respectively.

Parallel coordinates have been frequently used in visualizing many-objective solution sets. However, there have been some misinterpretations when parallel coordinates were used to claim the quality of solution sets. For example, a solution set has been claimed to have good convergence when it was seen within the range of the Pareto front in the parallel coordinates plot [27], [28], [29]. A solution set has been claimed to have good distribution when it was seen spreading over the whole range of the parallel coordinates plot [19], [28], [29], [30]. A solution set has been claimed to have poor diversity when it was seen concentrating in several polylines in the parallel coordinates plot [31]. A solution set has been claimed to have “noisy” distribution when it was seen cluttered in the parallel coordinates plot [20]. In next section, we will present what kind of quality aspects parallel coordinates can tell and what it cannot, along with examples to show the above claims misinterpreted (Figures 3, 9, 13(a) and 15(b), respectively).

## III. QUALITY MEASUREMENT

Given dimensionality reduction in the mapping of parallel coordinates, some loss of information is expected. In this section, we will see what and how much information parallel

coordinates can preserve and reflect in terms of the quality of a solution set in multi-objective optimization.

Often, the quality of a solution set in multi-objective optimization can be reflected via four measures: convergence, coverage, uniformity, and extensity. Convergence of a solution set measures the closeness of the set to the Pareto front; coverage considers the region of the set covering in comparison with the whole Pareto front; uniformity quantifies the distance between neighboring points in the set in the objective space; and extensity refers to the range of the set in the objective space. In general, there is no clear conceptual difference of these quality measures between many-objective optimization and multi-objective optimization with two or three objectives. However, many-objective optimization typically poses bigger challenge for evolutionary algorithms to achieve a good balance among these aspects.

A straightforward feature that parallel coordinates can tell is the range of a solution set. This feature can make it easy to interpret the extensity of a solution set, in comparison with the extensity metrics, e.g., *maximum spread* [32]. In the following, we will discuss if parallel coordinates can reflect other aspects of a solution set’s quality, i.e., convergence, coverage and uniformity.

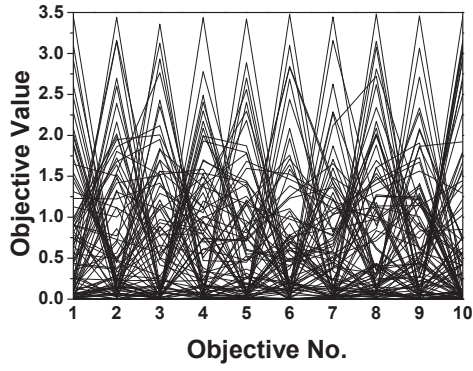
### A. Convergence

In multi-objective optimization, Pareto dominance is a fundamental criterion to compare solutions in terms of convergence. Parallel coordinates can clearly reflect the Pareto dominance relation between two solutions (such as polyline  $a$  being dominated by polyline  $b$  in Figure 1, assuming a minimization problem scenario) if the solution polylines are not overcrowded. It is worth mentioning that one can remove dominated solutions in parallel coordinates if they are only interested in non-dominated ones. This may make the plot clearer when comparing the quality of solution sets.

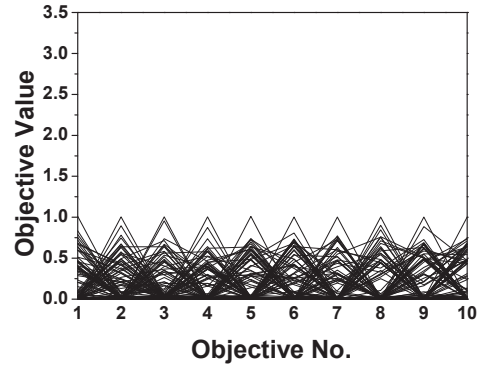
In addition to reflecting the Pareto dominance relation, parallel coordinates can largely imply the convergence of solution sets by their range. Figure 2 is such an example, where the parallel coordinates representation of two solution sets obtained by one run<sup>1</sup> of two EMO algorithms, NSGA-II [33] and GrEA [34], on the 10-objective DTLZ2 problem [35] is shown. As can be seen, NSGA-II has an inferior convergence, with its solution set ranging from 0 to around 3.5 in contrast to the problem’s Pareto front ranging from 0 to 1. GrEA has a good convergence on this problem and its solution set has the same range as the Pareto front. These observations can be confirmed by the results of the convergence metric  $GD^+$  [36] shown in the figure.  $GD^+$  is a modified version of the original  $GD$  [37], which makes it compatible with Pareto dominance.

However, we may not be able to accurately know the convergence of solution sets by their range shown in parallel

<sup>1</sup>The setting of the population size and maximum evaluations was 100 and 30,000, respectively. This setting was used in all conducted experiments in this paper, unless explicitly mentioned otherwise. In addition, the grid division in GrEA was set to 8.

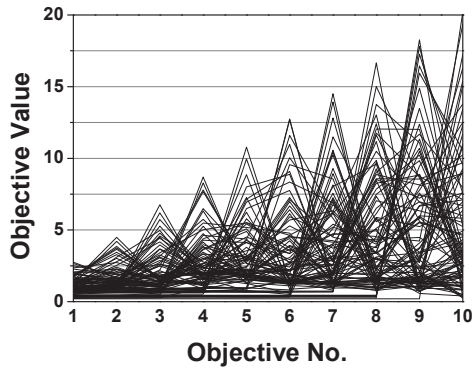


(a) NSGA-II ( $GD^+ = 2.26E-1$ )

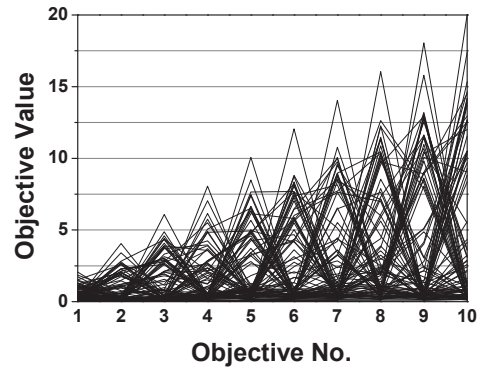


(b) GrEA ( $GD^+ = 1.20E-2$ )

Fig. 2. The solution sets obtained by NSGA-II and GrEA on the 10-objective DTLZ2, and their evaluation results on the convergence measure  $GD^+$  (the smaller the better).

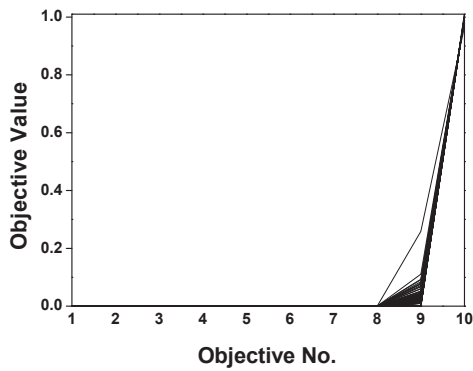


(a) NSGA-II ( $GD^+ = 3.63E-1$ )

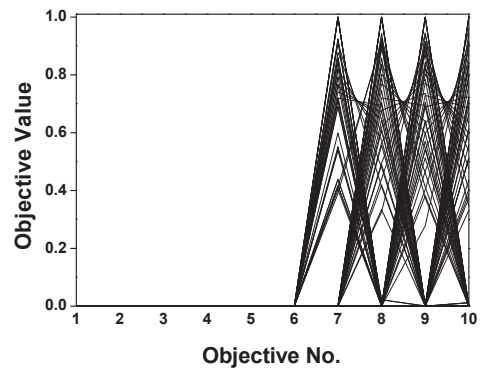


(b) GrEA ( $GD^+ = 6.32E-2$ )

Fig. 3. The solution sets obtained by NSGA-II and GrEA on the 10-objective WFG7, and their evaluation results on the convergence metric  $GD^+$  (the smaller the better).



(a) AR



(b) IBEA

Fig. 4. The solution sets obtained by AR and IBEA on the 10-objective DTLZ2.

coordinates. That is, even if two solution sets are located in the same range, they can perform considerably differently in terms of convergence. Figure 3 gives such an example, where solution sets obtained by one run of NSGA-II and GrEA on the 10-objective WFG7 problem [38] are shown. As seen, both algorithms virtually reach the range of the Pareto front (from 0 to  $2i$  where  $i$  is the objective index of the problem), but they have different  $GD^+$  results. NSGA-II is returned a significantly higher (worse)  $GD^+$  value than GrEA. This occurrence can be from two possibilities. One is that the solution set of NSGA-II is not actually close to the Pareto front. The other is that most of solutions in the set converge already while a small portion of the set is far away (but still in the range of the Pareto front).

In addition, it is worth mentioning that even if the “height” of two solution sets in the parallel coordinate plot is different, we may also not be able to tell the convergence difference between them if the range of a problem’s Pareto front is unknown. This is because different solution sets may converge into different parts of the Pareto front, especially in the situation where the Pareto front is highly convex.

### B. Coverage

In parallel coordinates, it is straightforward to see which region a solution set does not reach on any objective<sup>2</sup>. For example, in Figure 4 the solution set obtained by the AR method [39] concentrates in one tiny area and the set by IBEA [40] fails to cover the first six objectives on the 10-objective DTLZ2. Moreover, we can conjecture some distribution features of solution sets from their parallel coordinates representation. Take the solution sets in Figure 5 as an example; their parallel coordinates representation is shown in Figure 6. From Figure 6, we can know that the solution sets of IBEA and SMS-EMOA [41] fail to cover the region between 0.0 and 0.2 on all three objectives. Also, most of the solutions obtained by IBEA are located in the boundary of the Pareto front as there are very few lines distributed around the middle section on all three objectives in the figure.

However, there do exist some cases that different solution sets have the same parallel coordinates plots. We can easily construct such an example. In Figure 7, solution set B has a better coverage than set A (the four solutions in set A being duplicate), but the two sets have the same pattern in parallel coordinates (Figure 8). Note that if we change the order of some objectives (e.g.,  $f_1$  and  $f_2$ ), the parallel coordinates plots of the two solution sets in this example would be different.

One important fact that we would like to note is that as parallel coordinates map an  $m$ -dimensional graph onto a 2D graph they cannot fully reflect the coverage of solution sets. A set of solutions (represented by polylines) may have a good coverage over the range of the Pareto front in the 2D graph, but they may only cover part of the Pareto

front in the original  $m$ -dimensional space. An interesting example is shown in Figure 9. In that figure, NSGA-II appears to have a better coverage than GrEA according to the parallel coordinates plots, but GrEA has a better coverage evaluation result, measured by the coverage metric *Diversity Comparison Indicator* (DCI) [42].

### C. Uniformity

In parallel coordinates, it is not easy to see how evenly a set of solutions are distributed. However, a set of uniformly-distributed polylines in parallel coordinates often imply a uniformly-distributed solution set. As shown in Figures 10 and 11, MOEA/D [43] has a perfectly-distributed solution set and its corresponding polylines in parallel coordinates are distributed uniformly and regularly. This is in contrast to the solution set of NSGA-II which is distributed rather irregularly in both Cartesian and parallel coordinates plots. Note that a set of irregularly-distributed polylines may not represent a badly-distributed solution set, as uniformly-distributed solutions can have distinct values on different objectives. To show this, we select two EMO algorithms, MOEA/D [43] and BCE-MOEA/D [44], both of which are able to obtain a uniformly-distributed solution set on DTLZ2 (see Figure 12). In MOEA/D, the population distribution is maintained by a set of systematically-generated, uniformly-distributed weight vectors (within a simplex), and thus ideally its solutions only take several equivalent values on all the objectives. In contrast, in BCE-MOEA/D the population distribution is maintained by a niching-based criterion, and thus its solutions can spread over the whole range for each objective. Figure 13 gives the solution sets obtained by MOEA/D and BCE-MOEA/D on the 10-objective DTLZ2<sup>3</sup>. As seen, on the uniformity metric *Spacing* (SP)<sup>4</sup>[45], BCE-MOEA/D even performs better than MOEA/D, but we cannot see this from their parallel coordinates representation in the figure. This phenomenon may happen frequently when comparing decomposition-based algorithms having a set of systematically-generated weight vectors (such as MOEA/D and NSGA-III [46]) with algorithms that do not use such decomposition techniques (such as SPEA2 [47] and Two\_arch2 [48]). So care needs to be taken when making a conclusion about the distribution uniformity of solution sets from parallel coordinates.

Finally, it is worth mentioning that parallel coordinates plots can be easily cluttered with multiple lines overlaid. This may completely prevent solution sets’ distribution from being observed. Figures 14 and 15 show such an example, with two solution sets obtained by NSGA-II and SPEA2 on the 10-objective ML-DMP problem [49], [50]. The  $m$ -objective

<sup>3</sup>Here the number of DTLZ2’s decision variables is set to  $m - 1$  ( $m$  is the number of objectives) to ensure that all solutions produced by algorithms are Pareto optimal; thus the uniformity measure cannot be affected by the difference of solution sets’ convergence.

<sup>4</sup>In this paper, the SP metric has been slightly modified to make it compatible with Pareto dominance. That is, if two solution sets are comparable in terms of Pareto dominance, then the SP value of the dominating set is 0 and the SP value of the dominated set is 1.

<sup>2</sup>Note that for real-world problems whose Pareto front is unknown, we cannot tell if a solution set reaches the optimal region of objectives or not.

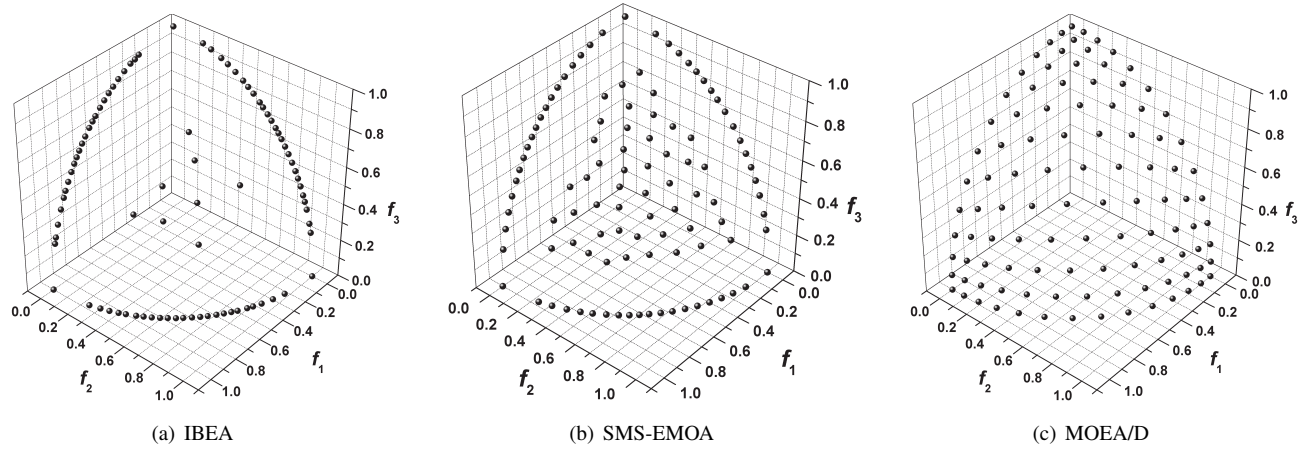


Fig. 5. The solution sets obtained by IBEA, SMS-EMOA and MOEA/D on the 3-objective DTLZ2, shown in Cartesian coordinates.

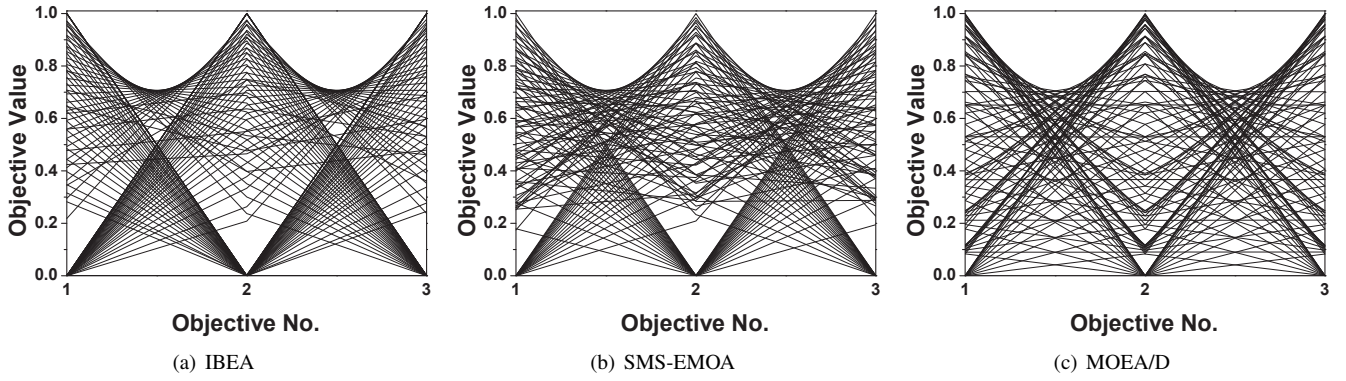


Fig. 6. The corresponding parallel coordinates of the solution sets in Figure 5.

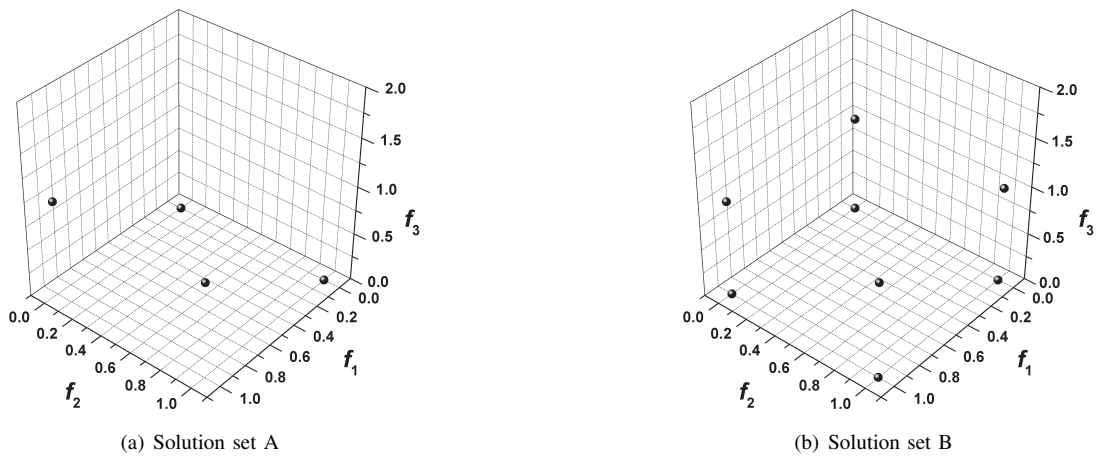
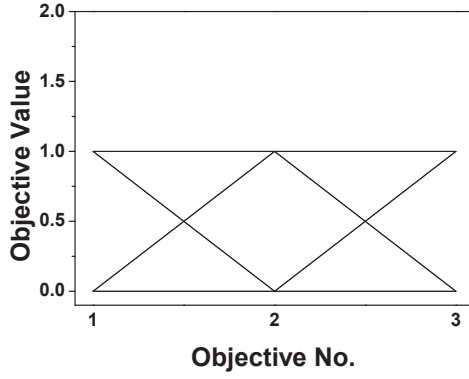
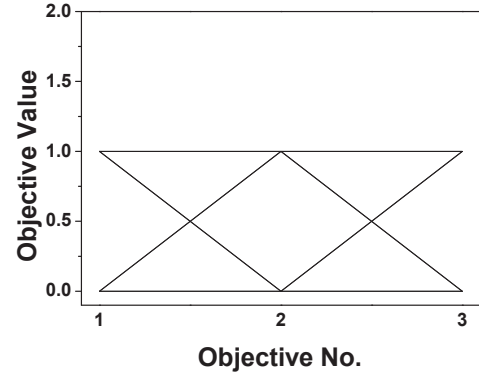


Fig. 7. An artificial example of two solution sets (A and B) having the same parallel coordinates plots shown in Figure 8.



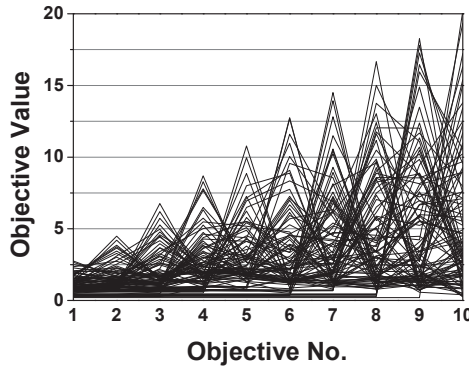


(a) Solution set A

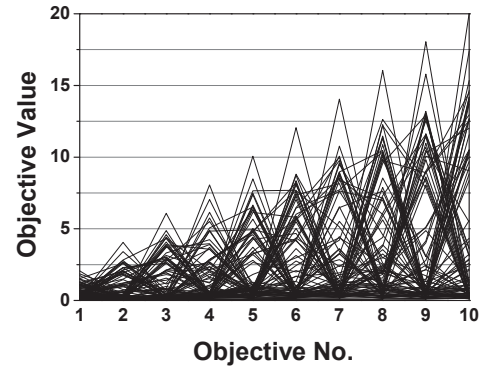


(b) Solution set B

Fig. 8. The parallel coordinates plots of the solution sets in Figure 7.

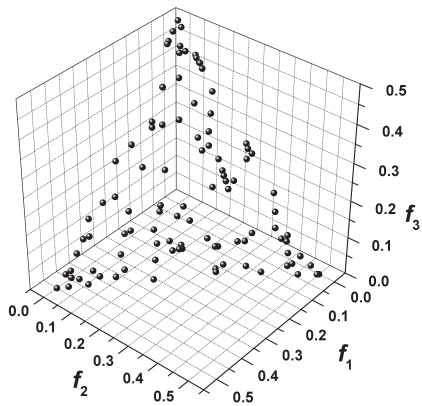


(a) NSGA-II (DCI = 7.17E-1)

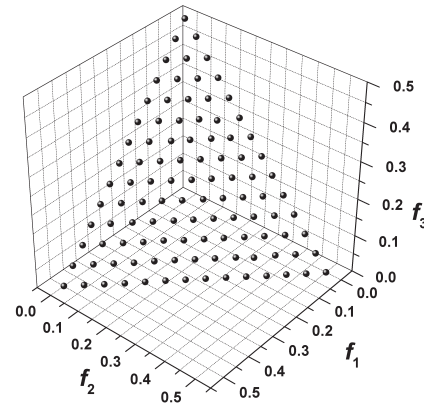


(b) GrEA (DCI = 7.75E-1)

Fig. 9. The solution sets obtained by NSGA-II and GrEA on the 10-objective WFG7, and their evaluation results on the coverage metric DCI (the bigger the better).

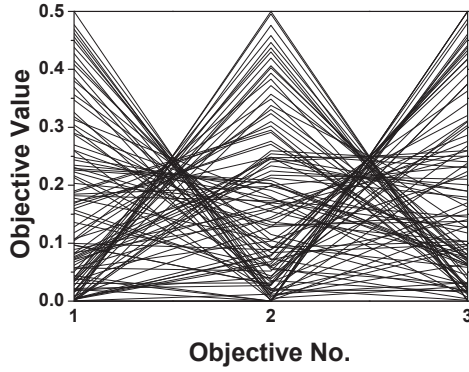


(a) NSGA-II

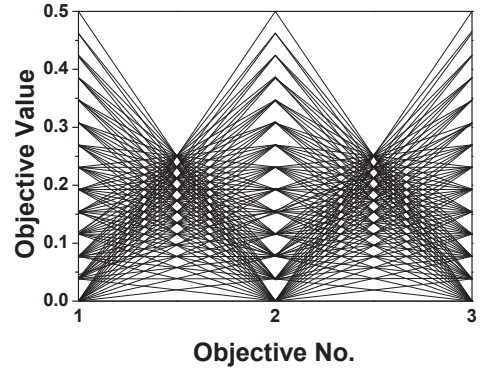


(b) MOEA/D

Fig. 10. The solution sets obtained by NSGA-II and MOEA/D on the 3-objective DTLZ1, shown in Cartesian coordinates.

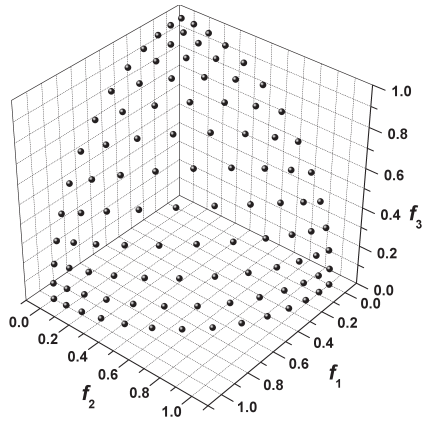


(a) NSGA-II

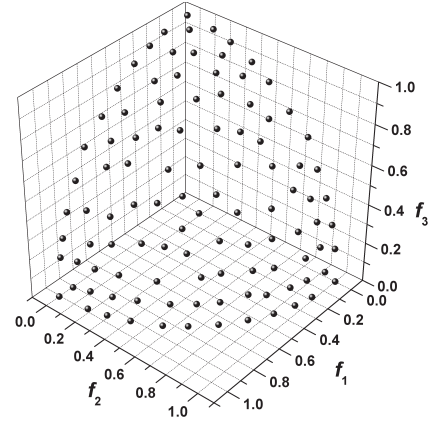


(b) MOEA/D

Fig. 11. The corresponding parallel coordinates plots of the solution sets in Figure 10.

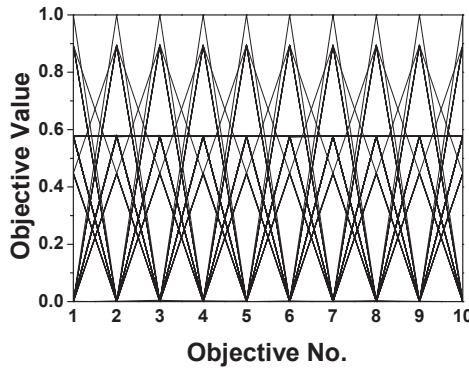


(a) MOEA/D

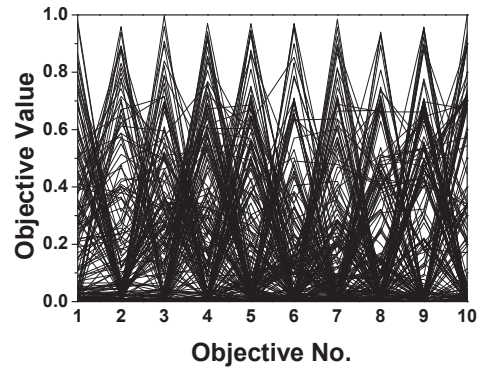


(b) BCE-MOEA/D

Fig. 12. The solution sets obtained by MOEA/D and BCE-MOEA/D on the 3-objective DTLZ2, shown in Cartesian coordinates.



(a) MOEA/D (SP = 1.05E-1)



(b) BCE-MOEA/D (SP = 7.74E-2)

Fig. 13. The solution sets obtained by MOEA/D and BCE-MOEA/D on the 10-objective DTLZ2, and their evaluation results on the uniformity metric SP (the smaller the better).



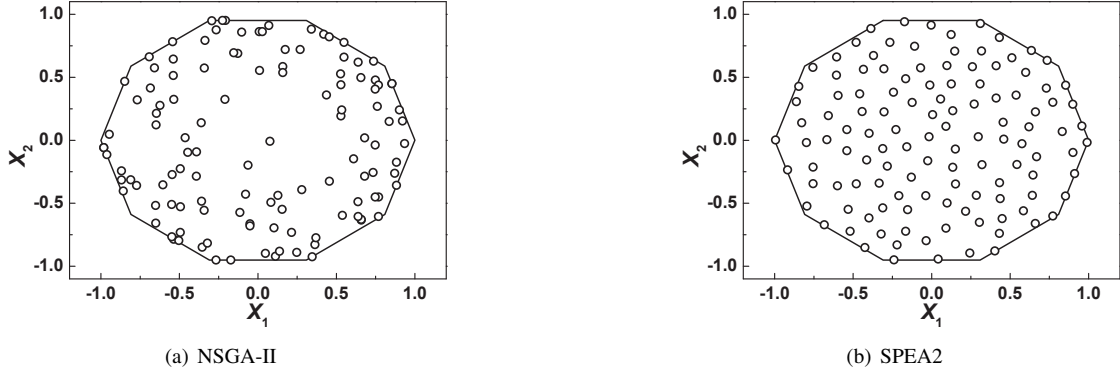


Fig. 14. The solution sets (in the decision space) obtained by NSGA-II and SPEA2 on the 10-objective ML-DMP where the search space is precisely the optimal polygon.

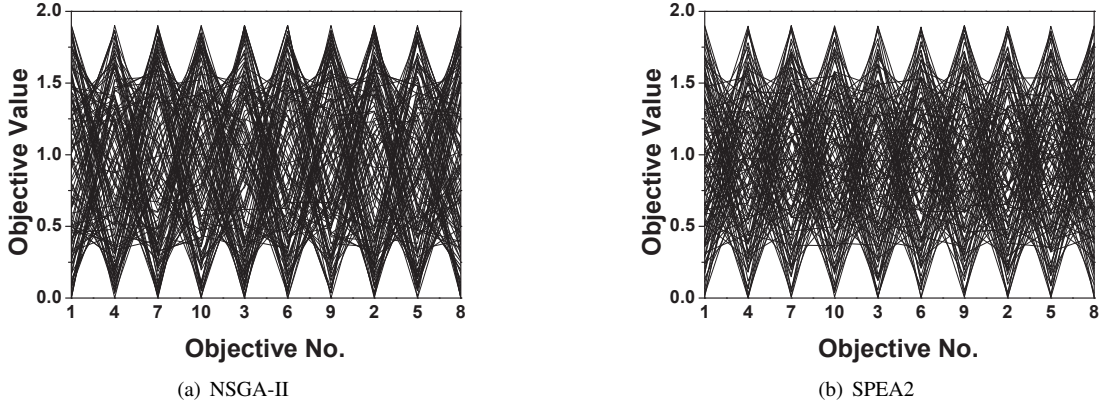


Fig. 15. The parallel coordinates plots of the solution sets (in the objective space) in Figure 14.

ML-DMP minimizes the distance of two-dimensional points to a set of  $m$  straight lines, each of which passes through one edge of a given regular polygon with  $m$  vertices. One interesting characteristic of ML-DMP is that the points in the regular polygon and their objective images are similar in the sense of Euclidean geometry. In other words, the ratio of the distance between any two points in the polygon to the distance between their corresponding objective vectors is a constant. This allows a straightforward understanding of the distribution of the objective vector set via observing the solution set in the 2D decision space. As can be seen in Figure 14, SPEA2 has a far better distribution uniformity than NSGA-II, but we cannot see the difference between their parallel coordinates representation in Figure 15.

#### IV. SOLUTION SET DISTRIBUTIONS

In parallel coordinates, it is straightforward to know the conflict between objectives. The number of intersection lines between adjacent objectives reflects their conflicting degree. If there is no intersection of any pair of lines between adjacent objectives, then these two objectives are completely non-conflicting (i.e., harmonious [51]), such as objectives  $f_1$  versus  $f_2$  and objectives  $f_2$  versus  $f_3$  in Figure 16. If there are many lines intersecting, then the two objectives

are heavily conflicting, such as objectives  $f_3$  versus  $f_4$  and objectives  $f_4$  versus  $f_5$  in Figure 16. If any pair of lines intersects, then the two objectives are completely conflicting to each other.

An interesting phenomenon in the parallel coordinates plot is that if all lines between two adjacent objectives intersect at one point, then these two objectives are negatively linearly dependent. Figure 17 is such an example. The four-objective ML-DMP problem minimizes the distance of points to four lines passing through the four edges of the given rectangle. From this definition, we can see that the two pairs of objectives,  $f_1$  versus  $f_3$  and  $f_2$  versus  $f_4$ , are negatively linearly dependent for the solutions in the rectangle ( $f_1 + f_3 = \sqrt{2}$ ,  $f_2 + f_4 = \sqrt{2}$ ). Therefore, each of the objective pairs intersects at one point, as shown in Figure 17(b).

This property is the known duality between the parallel coordinates representation and the Cartesian coordinate representation of data [1], [2]: points in Cartesian coordinates map into lines in parallel coordinates, while lines in Cartesian coordinates map into points in parallel coordinates. Take an example in [1], where a line  $\ell : f_2 = kf_1 + b$  in the Cartesian coordinate plane and two points lying on this line, say  $(x, kx + b)$  and  $(y, ky + b)$ , were considered (shown in

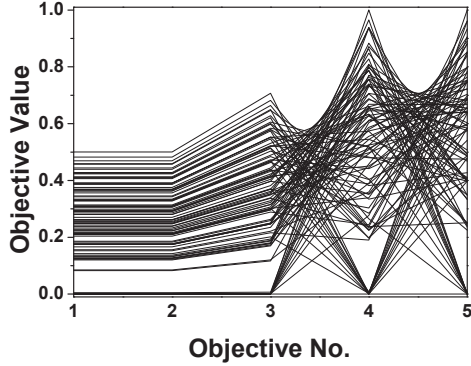


Fig. 16. The solution set obtained by SPEA2+SDE [52] on the 5-objective DTLZ5( $I, M$ ) [53], where  $I = 3$ .

Figure 18(a)). Figure 18(b) shows the corresponding parallel coordinates representation of the two points. For simplicity, let the distance between the vertical axes  $f_1$  and  $f_2$  be 1; then it is easy to know that the two lines intersect at a point given by  $\rho : ((1 - k)^{-1}, b(1 - k)^{-1})$  in parallel coordinates. This point depends only on  $k$  and  $b$ , the parameters of the original line in the Cartesian plane. This indicates that the parallel coordinates representation of any point on  $\ell$  passes through the point  $\rho$ .

From the coordinates of the point  $\rho$ , we can see the relation between the position of  $\rho$  and the slope  $k$  of the line  $\ell$ . If  $k < 0$ , the intersection occurs between the two parallel coordinates axes. Especially, when  $k = -1$  the intersection is precisely midway, as in the example of Figure 17(b). If  $1 < k$  or  $0 < k < 1$ , then the intersection point is on the left side or right side of the two coordinate axes, respectively. When  $k = \pm\infty$  or  $k = 0$ , the intersection point is on the left axis or right axis, respectively. Finally, when  $k = 1$ , the lines are parallel between the two axes in parallel coordinates. The above properties can help us understand the relation between objectives. For example, from the parallel coordinates plot in Figure 16 we know that  $f_1 = f_2$  and  $f_2 = kf_3 + b$ , where  $k > 1$  and  $b = 0$ .

Finally, note that since the horizontal position of the intersection point in parallel coordinates depends only on the slope  $k$ , when we can see many lines between two objectives in the parallel coordinates plot intersecting at the same horizontal position (but at different vertical positions), this means that many lines connecting two points in the Cartesian coordinate space have the same slope with respect to these two objectives. This occurs often when points in Cartesian coordinates are absolutely uniformly-distributed on the plane of these two objectives. The solution set in Figure 19 has such a pattern (see the midway of two adjacent objectives in Figure 19(b)). More interesting correspondence between the patterns of lines in parallel coordinates and the relation of objectives in the solution set can be found in [3].

## V. OBJECTIVE ORDER IN PARALLEL COORDINATES

In parallel coordinates, each axis has at most two neighboring axes (one on the left and one on the right). Different order of objective axes presents different information with respect to the relation between objectives. Take Figure 20 as an example. In Figure 20(a) where the order of objectives is  $f_1, f_2, f_3, f_4, f_5$ , the conflict between any two adjacent objectives is rather weak. In contrast, in Figure 20(b) where the order of objectives is  $f_1, f_3, f_5, f_2, f_4$ , the conflict between any two adjacent objectives is quite intense.

In a solution set with  $m$  objectives, its parallel coordinates representation can only show  $m - 1$  relationships at a time. This can be a very small portion compared to the total  $\binom{m}{2}$  relationships existing in  $m$  objectives. Therefore, a good objective axis arrangement providing the user as much (clear) information as possible is of importance. As shown in Figures 21 and 22, after swapping some objectives, we can see interesting patterns (linearly dependent) between some pairs of objectives. Similar observations have been reported by Freitas et al. [54]. In [54], the authors also proposed an objective axis-rearranging method by placing the most harmonious objectives in a row in many-objective optimization. However, this rearrangement may not be able to present the information of objectives being severely in conflict (e.g., negatively linearly dependent). In fact, determining a good order of the axes in the parallel coordinates plot (to reflect as much as useful information) is nontrivial. There exist some work in the data visualization field, e.g., methods to reduce clutter in the parallel coordinates plot [55], [56].

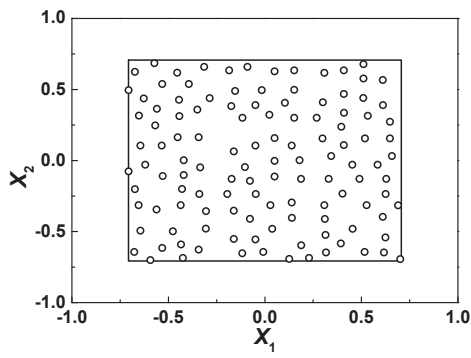
## VI. HOW TO DRAW A PARALLEL COORDINATE PLOT

In this section, we give procedures of how to plot a solution set in parallel coordinates by several commonly-used graphing tools: MS Excel, MATLAB, LaTeX, and Origin. Tables I–IV provide the steps/codes by MS Excel, MATLAB, LaTeX, and Origin, respectively. Here, we use the example in Figure 1. Figure 23 also presents the graphs drawn by the four tools to that example. Finally, we would like to note that in this paper all of the parallel coordinates graphs of data examples were drawn by Origin, and these data examples are available at <http://www.cs.bham.ac.uk/%7Eelimx>.

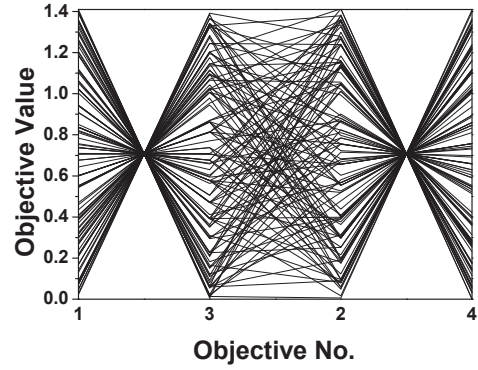
## VII. CONCLUSIONS

The Parallel coordinates plot has drawn increasing attention in many-objective optimization, but mapping a many-objective solution set onto a 2D parallel coordinates plane may not be straightforward to reveal the information contained in the set. This paper has made some observations on the use of the parallel coordinates plot to present a solution set in many-objective optimization. In particular,

- The parallel coordinates representation of a solution set can *partly* reflect its convergence, coverage and uniformity. This suggests that the parallel coordinates plot can be an assistant tool (but not entirely replacing quality metrics) in assessing a many-objective solution set.

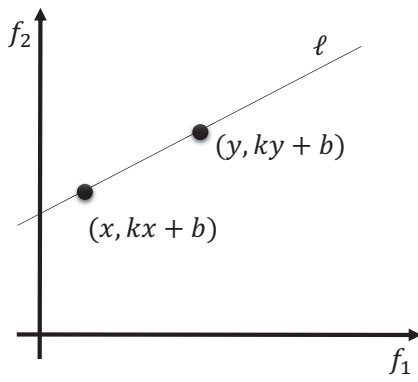


(a) Decision space

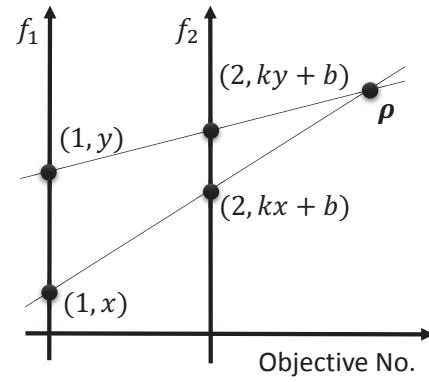


(b) Objective space

Fig. 17. The solution set of SPEA2+SDE on the 4-objective ML-DMP.

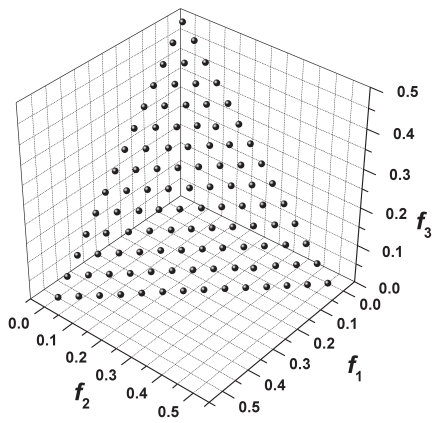


(a) Cartesian coordinates

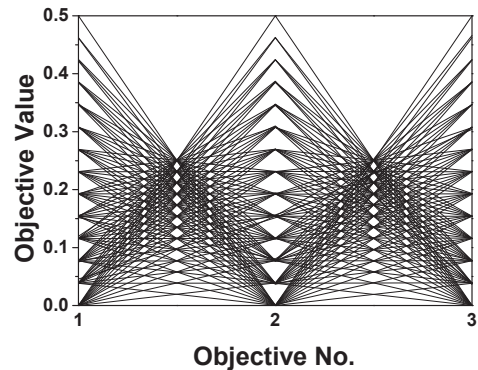


(b) Parallel coordinates

Fig. 18. An example of a line  $\ell : f_2 = kf_1 + b$  in the Cartesian coordinate plane corresponding to a point  $\rho : ((1 - k)^{-1}, b(1 - k)^{-1})$  in the parallel coordinates plane.



(a) Cartesian coordinates



(b) Parallel coordinates

Fig. 19. The solution set obtained by MOEA/D on the 3-objective DTLZ1.

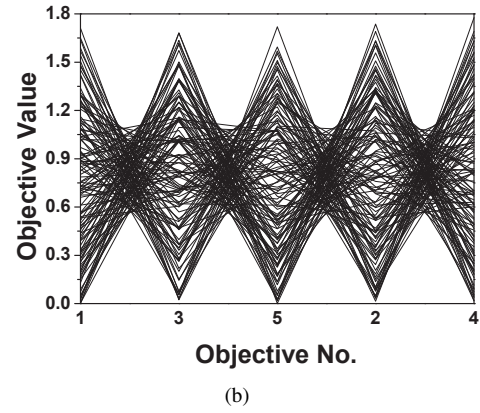
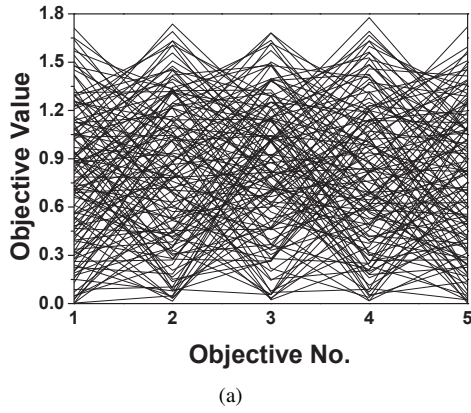


Fig. 20. The solution set of SPEA2+SDE on the 5-objective ML-DMP, shown by different order of objectives in parallel coordinates.

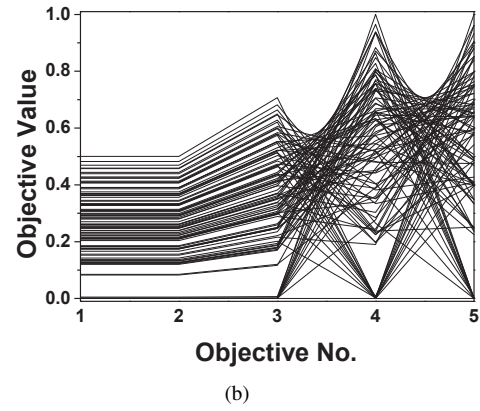
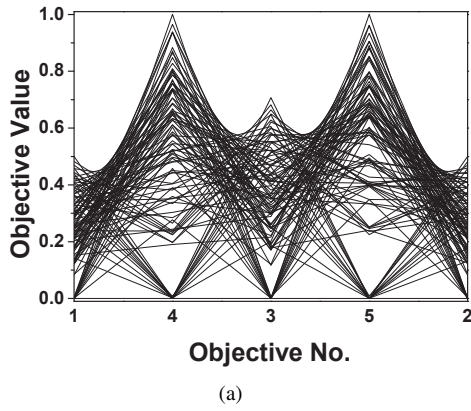


Fig. 21. The solution set of SPEA2+SDE on the 5-objective DTLZ5(I,M) where  $I = 3$ , shown by different order of objectives in parallel coordinates.

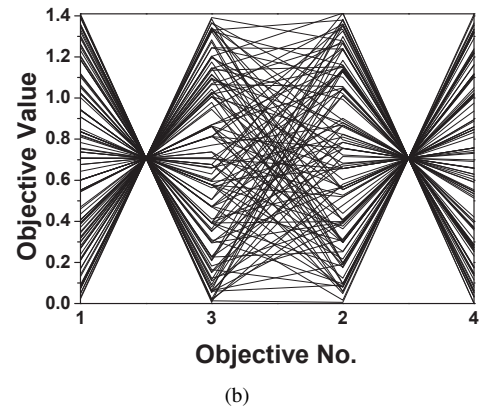
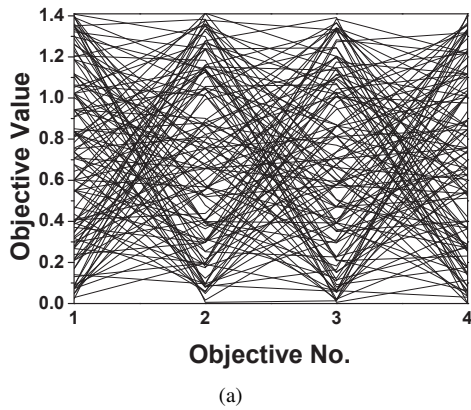


Fig. 22. The solution set of SPEA2+SDE on the 4-objective ML-DMP, shown by different order of objectives in parallel coordinates.



TABLE I

STEPS OF CREATING A PARALLEL COORDINATES PLOT IN MS EXCEL.

- (1) Input the data as a table with each row as a solution and select them.
- (2) Click Insert -> Recommended Charts.
- (3) On the Recommended Charts tab, scroll through the list of charts that Excel recommends for your data, click Line chart -> OK.
- (4) Use the Chart Elements, Chart Styles, and Chart Filters buttons next to the upper-right corner of the chart to add chart elements like axis titles or data labels, customize the look of your chart, or change the data shown in the chart.

TABLE II

CODES OF CREATING A PARALLEL COORDINATES PLOT IN MATLAB.

```
X = [15 31 20 50; 10 18 2 30; 20 5 32 20];
groups = {'a', 'b', 'c'};
parallelcoords(X,'group',groups);
xlabel('Objective No.');
```

TABLE III

INSTRUCTIONS FOR PLOTTING PARALLEL COORDINATES IN LATEX USING THE PGFPLOTS PACKAGE.

- (1) Include the pgfplots package by adding the following line to your preamble:  
`\usepackage{pgfplots}`
- (2) Plot with the following commands:  
`\begin{tikzpicture}`  
`\begin{axis}[xlabel={Objective No.},`  
`ylabel={Objective Value}, xtick=data,`  
`symbolic x coords={1, 2, 3, 4}]`  
`\addplot+[mark=none,draw=black,sharp plot]`  
`plot coordinates {(1,15) (2,31) (3,20) (4,50)};`  
`\addplot+[mark=none,draw=red,sharp plot]`  
`plot coordinates {(1,10) (2,18) (3,2) (4,30)};`  
`\addplot+[mark=none,draw=blue,sharp plot]`  
`plot coordinates {(1,20) (2,5) (3,32) (4,20)};`  
`\end{axis}`  
`\end{tikzpicture}`

TABLE IV

STEPS OF CREATING A PARALLEL COORDINATES PLOT IN ORIGIN.

- (1) Create a table consisting of the first column being X axis from 1 to  $m$  (where  $m$  is the number of objectives) and the remaining columns being Y axis with each column for a solution.
- (2) Select the table and click the Line button at the lower-left corner of the panel.
- (3) Double click the Axis Labels and the polylines of parallel coordinates to customize the look of the chart.

- Although the clarity can be affected by overlapping polygonal lines, parallel coordinates transform certain geometrical features of a many-objective solution set into easily seen 2D patterns.
- The order of objective axes matters in parallel coordinates. To better present the relationship between objectives, it may need to be rearranged according to features of the solution set at hand.

Our subsequent study is towards overcoming/alleviating the difficulties of interpreting the parallel coordinates plot presented in this paper. Particularly, how to arrange the order of objectives will be the focus of our future work as it had presented its usefulness in the paper. In this regard, a straightforward thought is to place the most conflicting objectives or the most harmonious objectives together so that people could see some meaningful patterns (such as the examples in Figures 21 and 22). Another thought is to consider the coverage of the lines between objectives in a parallel coordinates plot; people may acquire more information from less coverage of the lines, for example, after exchanging the order of objectives  $f_1$  and  $f_2$  in Figure 8.

## ACKNOWLEDGEMENT

The authors would like to thank the Associate Editor and the three reviewers for their thoughtful suggestions and constructive comments. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) of U.K. under Grants EP/K001523/1 and EP/J017515/1, and National Natural Science Foundation of China (NSFC) under Grants 61329302 and 61403326. X. Yao was also supported by a Royal Society Wolfson Research Merit Award.

## REFERENCES

- [1] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 2, pp. 69–91, August 1985.
- [2] E. J. Wegman, "Hyperdimensional data analysis using parallel coordinates," *Journal of the American Statistical Association*, vol. 85, no. 411, pp. 664–675, September 1990.
- [3] A. Inselberg, *Parallel coordinates*. Boston, MA: Springer US, 2009, pp. 2018–2024.
- [4] K. Miettinen, "Survey of methods to visualize alternatives in multiple criteria decision making problems," *OR Spectrum*, vol. 36, no. 1, pp. 3–37, January 2014.



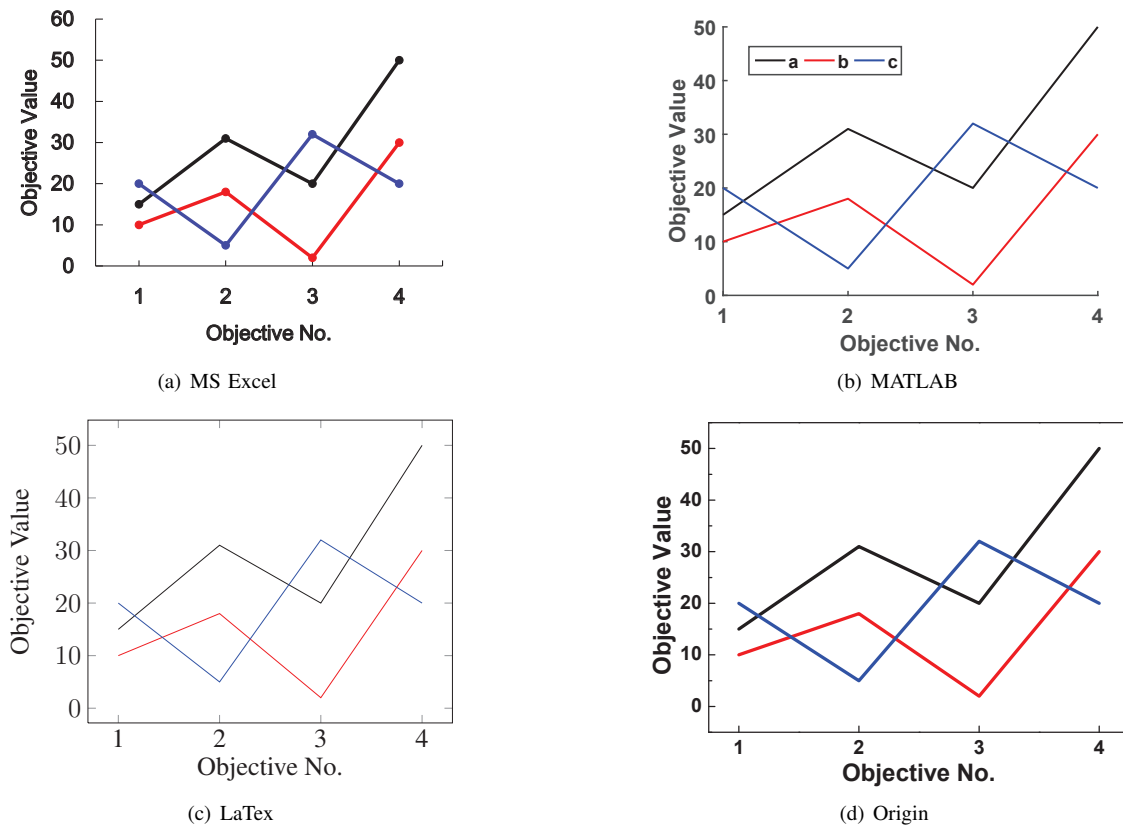


Fig. 23. Examples of the parallel coordinates plot in MS Excel, MATLAB, LaTeX, and Origin.

- [5] C. M. Fonseca and P. J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation," *IEEE Trans. on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 28, no. 1, pp. 26–37, January 1998.
- [6] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc., 2001.
- [7] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *Proc. of 2008 IEEE Congress on Evolutionary Computation*, Hong Kong, China, June 2008, pp. 2419–2426.
- [8] R. C. Purshouse and P. J. Fleming, "Evolutionary many-objective optimisation: An exploratory analysis," in *Proc. of 2003 IEEE Congress on Evolutionary Computation*, vol. 3, Canberra, ACT, Australia, December 2003, pp. 2066–2073.
- [9] P. Fleming, R. Purshouse, and R. Lygoe, "Many-objective optimization: An engineering design perspective," in *Proc. of 3rd International Conference on Evolutionary Multi-Criterion Optimization*, Guanajuato, Mexico, March 2005, pp. 14–32.
- [10] D. J. Walker, R. Everson, and J. E. Fieldsend, "Visualizing mutually nondominating solution sets in many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 17, no. 2, pp. 165–184, April 2013.
- [11] T. Tuar and B. Filipi, "Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the projection method," *IEEE Trans. on Evolutionary Computation*, vol. 19, no. 2, pp. 225–245, April 2015.
- [12] Z. He and G. G. Yen, "Visualization and performance metric in many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 3, pp. 386–402, June 2016.
- [13] T. Wagner, N. Beume, and B. Naujoks, "Pareto-, aggregation-, and indicator-based methods in many-objective optimization," in *Proc. of 4th International Conference on Evolutionary Multi-Criterion Optimization*, Matsushima, Japan, March 2007, pp. 742–756.
- [14] H. K. Singh, A. Isaacs, and T. Ray, "A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems," *IEEE Trans. on Evolutionary Computation*, vol. 15, no. 4, pp. 539–556, August 2011.
- [15] M. Li, S. Yang, X. Liu, and R. Shen, "A comparative study on evolutionary algorithms for many-objective optimization," in *Proc. of 7th International Conference on Evolutionary Multi-Criterion Optimization*, Sheffield, UK, March 2013, pp. 261–275.
- [16] U. K. Wickramasinghe and X. Li, "Using a distance metric to guide PSO algorithms for many-objective optimization," in *Proc. of 2009 Conference on Genetic and Evolutionary Computation*, Montreal, Quebec, Canada, July 2009, pp. 667–674.
- [17] R. J. Lygoe, M. Cary, and P. J. Fleming, "A many-objective optimisation decision-making process applied to automotive diesel engine calibration," in *Proc. of 8th International Conference on Simulated Evolution and Learning*, Kanpur, India, December 2010, pp. 638–646.
- [18] R. Wang, R. C. Purshouse, and P. J. Fleming, "“Whatever Works Best for You” - A new method for a priori and progressive multi-objective optimisation," in *Proc. of 7th International Conference on Evolutionary Multi-Criterion Optimization*, Sheffield, UK, March 2013, pp. 337–351.
- [19] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, February 2016.
- [20] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, October 2016.
- [21] Y. Xiang, Y. Zhou, M. Li, and Z. Chen, "A vector angle-based evolutionary algorithm for unconstrained many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 21, no. 1, pp. 131–152, February 2017.
- [22] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Difficulties in specifying reference points to calculate the inverted generational distance for many-objective optimization problems," in *Proc. of 2014 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making*, Orlando, FL, USA, December 2014, pp. 170–177.

- [23] M. Li, S. Yang, and X. Liu, "A performance comparison indicator for Pareto front approximations in many-objective optimization," in *Proc. of 2015 Conference on Genetic and Evolutionary Computation*, Madrid, Spain, July 2015, pp. 703–710.
- [24] H. Wang, Y. Jin, and X. Yao, "Diversity assessment in many-objective optimization," *IEEE Trans. on Cybernetics*, vol. 47, no. 6, pp. 1510–1522, June 2017.
- [25] W. Hu and G. G. Yen, "Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system," *IEEE Trans. on Evolutionary Computation*, vol. 19, no. 1, pp. 1–18, February 2015.
- [26] R. Hernández Gómez, C. A. Coello Coello, and E. Alba Torres, "A multi-objective evolutionary algorithm based on parallel coordinates," in *Proc. of 2016 Genetic and Evolutionary Computation Conference*, Denver, Colorado, USA, July 2016.
- [27] J. Cheng, G. Yen, and G. Zhang, "A many-objective evolutionary algorithm with enhanced mating and environmental selections," *IEEE Trans. on Evolutionary Computation*, vol. 19, no. 4, pp. 592–605, August 2015.
- [28] M. Li, S. Yang, and X. Liu, "Bi-goal evolution for many-objective optimization problems," *Artificial Intelligence*, vol. 228, pp. 45–65, November 2015.
- [29] X. Bi and C. Wang, "A many-objective evolutionary algorithm based on hyperplane projection and penalty distance selection," *Natural Computing*, to be published.
- [30] Z. Li, L. Zhang, Y. Su, J. Li, and X. Wang, "A skin membrane-driven membrane algorithm for many-objective optimization," *Neural Computing & Applications*, to be published.
- [31] L. Cai, S. Qu, Y. Yuan, and X. Yao, "A clustering-ranking method for many-objective optimization," *Applied Soft Computing*, vol. 35, pp. 681–694, October 2015.
- [32] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, June 2000.
- [33] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, April 2002.
- [34] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, October 2013.
- [35] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, A. Abraham, L. Jain, and R. Goldberg, Eds. London: Springer London, 2005, pp. 105–145.
- [36] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Modified distance calculation in generational distance and inverted generational distance," in *Proc. of 8th International Conference on Evolutionary Multi-Criterion Optimization*, Guimarães, Portugal, March 2015, pp. 110–125.
- [37] D. A. V. Veldhuizen and G. B. Lamont, "Evolutionary computation and convergence to a Pareto front," in *Late Breaking Papers at the Genetic Programming Conference*. Morgan Kaufmann, 1998, pp. 221–228.
- [38] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, October 2006.
- [39] P. J. Bentley and J. P. Wakefield, *Finding acceptable solutions in the Pareto-optimal range using multiobjective genetic algorithms*. London: Springer-Verlag London, 1998, pp. 231–240.
- [40] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. of 8th International Conference on Parallel Problem Solving from Nature*, Birmingham, UK, September 2004, pp. 832–842.
- [41] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, September 2007.
- [42] M. Li, S. Yang, and X. Liu, "Diversity comparison of Pareto front approximations in many-objective optimization," *IEEE Trans. on Cybernetics*, vol. 44, no. 12, pp. 2568–2584, December 2014.
- [43] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, December 2007.
- [44] M. Li, S. Yang, and X. Liu, "Pareto or non-Pareto: Bi-criterion evolution in multi-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 5, pp. 645–665, October 2016.
- [45] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1995.
- [46] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, August 2014.
- [47] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design, Optimisation, and Control*. Barcelona, Spain: CIMNE, 2002, pp. 95–100.
- [48] H. Wang, L. Jiao, and X. Yao, "TwoArch2: An improved two-archive algorithm for many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 19, no. 4, pp. 524–541, August 2015.
- [49] M. Li, S. Yang, and X. Liu, "A test problem for visual investigation of high-dimensional multi-objective search," in *Proc. of 2014 IEEE Congress on Evolutionary Computation*, Beijing, China, July 2014, pp. 2140–2147.
- [50] M. Li, C. Grosan, S. Yang, X. Liu, and X. Yao, "Multi-line distance minimization: A visualized many-objective test problem suite," *IEEE Trans. on Evolutionary Computation*, to be published.
- [51] R. C. Purshouse and P. J. Fleming, "Conflict, harmony, and independence: Relationships in evolutionary multi-criterion optimisation," in *Proc. of 2nd International Conference on Evolutionary Multi-Criterion Optimization*, Faro, Portugal, April 2003, pp. 16–30.
- [52] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for Pareto-based algorithms in many-objective optimization," *IEEE Trans. on Evolutionary Computation*, vol. 18, no. 3, pp. 348–365, June 2014.
- [53] K. Deb and D. K. Saxena, "Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems," in *Proc. of 2006 IEEE Congress on Evolutionary Computation*, Vancouver, BC, Canada, July 2006, pp. 3352–3360.
- [54] A. R. de Freitas, P. J. Fleming, and F. G. Guimares, "Aggregation Trees for visualization and dimension reduction in many-objective optimization," *Information Sciences*, vol. 298, pp. 288–314, March 2015.
- [55] G. Ellis and A. Dix, "A taxonomy of clutter reduction for information visualisation," *IEEE Trans. on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1216–1223, November 2007.
- [56] L. F. Lu, M. L. Huang, and J. Zhang, "Two axes re-ordering methods in parallel coordinates plots," *Journal of Visual Languages & Computing*, vol. 33, pp. 3–12, April 2016.