Honda Research Institute EU
**Honda Research Institute Europe GmbH**
https://www.honda-ri.de/

# A Method for a Posteriori Identification of Knee Points Based on Solution Density

## Guo Yu, Yaochu Jin, Markus Olhofer

### 2018

**Preprint:**

This is an accepted article published in 2018 World Congress on Computational Intelligence. The final authenticated version is available online at: https://doi.org/[DOI not available]

# A Method for *a Posteriori* Identification of Knee Points Based on Solution Density

Guo Yu
*Department of Computer Science*
*University of Surrey*
Guildford, Surrey, GU2 7XH, UK.
guo.yu@surrey.ac.uk

Yaochu Jin
*Department of Computer Science*
*University of Surrey*
Guildford, Surrey, GU2 7XH, UK.
yaochu.jin@surrey.ac.uk

Markus Olhofer
*Honda Research Institute Europe GmbH*
*Carl-Legien-Strasse 30*
D-63073 Offenbach/Main, Germany.
markus.olhofer@honda-ri.de

***Abstract*—Many evolutionary algorithms have been proposed and demonstrated to have excellent performance in striking a balance between convergence and diversity in dealing with multi-objective optimization problems. However, little attention has been paid to the decision making stage where a small number of solutions are selected to be presented to the user. It is believed that knee points are considered to be the naturally preferred solutions when no specific preferences are available, because knee solutions incur a large loss in at least one objective to gain a small amount in other objectives. One common issue in the identification of knee points is that some knee points are easily ignored and knees in concave regions are hard to be identified. To resolve these issues, this paper proposes a novel method for knee identification, which first maps the non-dominated solutions to a constructed hyperplane and then divides them into groups, each representing a candidate knee region, based on the density of the solutions projected on the hyperplane. Finally, the convexity and curvature of the candidate knee groups are determined and only those having a strong curvature are kept. The proposed method is empirically demonstrated to be effective in identifying knee points located in both convex and concave regions on three existing test problems and one newly proposed test problem.**

## I. Introduction

Without loss of generality, consider the following multi-objective minimization problem:

$$\min_{\overrightarrow{x}} : F(\overrightarrow{x}) = (f_1(\overrightarrow{x}), \cdots, f_m(\overrightarrow{x})), \tag{1}$$

where $\overrightarrow{x} = (x_1, \cdots, x_n) \in \mathbb{X}$ is the decision vector. $\mathbb{X} \subseteq \mathbb{R}^n$ is the decision space, and $n$ is the number of decision variables. $F : \mathbb{X} \to \mathbb{R}^m$ consists of $m$ objectives.

In the past decades, many multiobjective evolutionary algorithms (MOEAs) have been developed and demonstrated to be effective in dealing with MOPs in terms of being able to achieve a good balance between convergence and diversity of the obtained tradeoff solutions. MOEAs can be roughly classified into four categories, namely, dominance-based methods [1]–[4], indicators or metrics based approaches [5]–[7], aggregation or decomposition based methods [8]–[11], as well as preference-based methods [12]–[15].

Although a set of well-distributed and well-converged solutions can be obtained, it is hard for the decision makers (DMs) to select a small number of interested solutions from the solution set. To address this issue preference information from the DM needs to be introduced into the optimization process, leading to a more focused search. Preference-based MOEAs can largely be classified into three categories according to the time to integrate the preference [16]–[18]: *a priori*, interactive, and *a posteriori* approaches. One common implicit assumption in preference-based MOEAs is that the DM has certain *a priori* knowledge of the optimization problem.

Thus, how to select solutions from the large number of non-dominated solutions achieved by an MOEA remains an open problem in case no specific preferences are available. It is widely believed that knee points are naturally preferred solutions, since a large compromise in at least one objective is required to gain a small amount in other objectives [19], [20]. Moreover, knee points are sometimes located in the middle of the Pareto front, contribute to a large dominated hypervolume [21]. According to [22], [23], knee points can be categorized into convex, concave, and edge knee points.

Several methods for identifying knee points have been reported in the literature. Das et al. [24], [25] use the maximum distance to a constructed hyperplane to identify the knees in the convex regions, which, however, may miss many local knee points. Branke et al. [26] construct the expected marginal utility (EMU) with a number of uniform weight vectors, and knees will be identified with the largest EMU values. This method, unfortunately, may also fail to identify some local knee points in the convex regions. Its extension [23] aims to create a complete ordering of the solutions and recursively uses the EMU to identify knee points. Unfortunately the method is not well suited for identifying knees in concave regions of a Pareto front (PoF). The reflex/bend angle [26], [27] describes knee points with the largest reflex angle, and a variant proposed by Deb and Gupta [22] slightly modifies the description of the bend angle by giving a pair of values $(\alpha > 1, \beta > 1)$. Note that these two methods are limited to two-objective optimization problems. The method proposed in [28], [29] describes the knees by means of the ratio between the improvement and deterioration when the objectives of two solutions are exchanged. Similarly, the method may miss the knee points in concave regions of the PoF.

To resolve the aforementioned common issues in existing algorithms, this paper proposes a new knee identification algorithm by estimating the density of solutions projected

onto the hyperplane constructed by the extreme points of the non-dominated solutions set. Based on the solution density, candidate knee regions will be identified and only those having a large curvature estimated on the basis of the radial coordinate value suggested in [30] are retained. Finally, a small number of solutions near each candidate knee point will be presented to the DM together with the knees.

The rest of the paper is organized as follows. Section II provides a brief introduction of definitions related to Pareto dominance and knee points. Section III describes the proposed knee point identification method in detail. The comparative experiments are presented in Section IV. Section V concludes the paper.

## II. Parero dominance and knee points

In dealing with the MOP defined in (1), the following definitions are needed.

**Definition 1.** Given two solutions $x, y \in \mathbb{X}$, solution $x$ is defined to Pareto dominate $y$, which denotes as $x \prec y$, if $\forall i \in \{1, 2, \cdots, m\}$, $f_i(x) \leq f_i(y)$ and $\exists j \in \{1, 2, \cdots, m\}, f_j(x) < f_j(y)$.

**Definition 2.** A solution $x \in \mathbb{X}$ is nondominated if $\nexists y \in \mathbb{X}$, $s.t. f_i(y) \leq f_i(x)$, $\forall i \in \{1, 2, \cdots, m\}$.

**Definition 3.** A solution $x \in \mathbb{X}$, is said to weakly dominate $y$, if $\forall i \in \{1, 2, \cdots, m\}$, $f_i(x) \leq f_i(y)$.

**Definition 4.** The set $X \subset \mathbb{X}$ of all the nondominated solutions is called the nondominated or Pareto-optimal set (PoS). The Pareto-optimal front (PoF) is the image of PoS in the objective space.

A knee point, visually located at the convex or concave bulge on a PoF, requires a large compromise in at least one objective to gain a small amount in others. As stated in [24], it is empirically noticed that the user or designer usually intends to select a solution in the middle of the surface, which most likely is a knee point. In [22], [23], Deb et al. have provided illustrations of different types of knee points and classify them into convex, concave, and edge knee points. For example in Fig. 1, $B$ is located in the concave region of the PoF and therefore is a concave knee point. Similarly, $C$ and $F$ are convex knees, and $D$ and $E$ are edge knees of the disconnected fragments of the PoF, respectively. $A$ and $I$ are extreme points.

## III. Proposed knee identification method

The proposed algorithm in pseudo code is described in Algorithm 1, which consists of two main steps, transformation and identification. In the following, we present in greater detail the main components in each step.

### A. Transformation

The transformation stage linearly maps all non-dominated solutions onto a constructed hyperplane. Before transformation, solutions are normalized so that they are all in the first quadrant. Then a hyperplane is constructed using the extreme solutions of the non-dominated set. Finally, all solutions are mapped onto the constructed hyperplane.

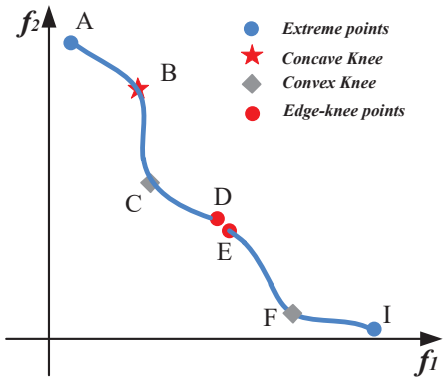



Fig. 1. Different kinds of interesting points.

**Algorithm 1** The pseudo-code of the proposed knee point identification method

**Input:** Representative population: $P$
**Output:** Groups of Knees: $K^{'}$

$Transformation$ :
- $Q = Normalize(P)$
- $S = Mapping(Q)$

$Identification$ :
**while** $S! = \emptyset$ **do**
  $DensityEstimation(S)$
  $K = Distinguish(S)$
  $K^{'} = Sort(K)$
  $S = S/K^{'}$
**end while**
$Output(K^{'})$

Let $u^i_{min} = \min_{j=1}^{|P|} f_i(x_j)$ and $u^i_{max} = \max_{j=1}^{|P|} f_i(x_j)$ are ideal and nadir points of the $i$th objective, respectively, the $i$th objective of solution $x$ can be normalized as follows:

$$f_i^{'}(x) = \frac{f_i(x) - u^i_{min}}{u^i_{max} - u^i_{min}} \quad \text{for } i = 1, \cdots, m \tag{2}$$

, where $m$ is the number of objectives.

Given a normalized solution $p = (f_1^{'}, \cdots, f_m^{'})$ and hyperplane $\beta : f_1 + \cdots + f_m = 1$, where $m$ is the number of objectives, the normal vector can be characterized by $\overrightarrow{n} = (1, 1, \cdots, 1)$ or $\overrightarrow{n} = (-1, -1, \cdots, -1)$ and $|\overrightarrow{n}| = \sqrt{m}$. Then the mapping point $p^{'} = (p_1, \cdots, p_m)$ can be calculated as follows:

$$\overrightarrow{pp^{'}} = |\overrightarrow{pp^{'}}| \cdot \frac{\overrightarrow{n}}{|\overrightarrow{n}|} \tag{3}$$

, where $\overrightarrow{pp^{'}}$ denotes a vector and $|\cdot|$ the norm of the vector.

Thus, the signed distance of solution $p$ to a hyperplane,

$|\overrightarrow{pp'}|$, can be calculated as follows:

$$\begin{cases} \beta: & f_1 + \cdots + f_m - 1 = 0 \\ |\overrightarrow{pp'}| & = \frac{f_1' + \cdots + f_m' - 1}{|\overrightarrow{n}|} \end{cases} \tag{4}$$

. Note that $|\overrightarrow{pp'}| < 0$ means point $p$ is below the hyperplane $\beta$, and $|\overrightarrow{pp'}| > 0$ means that $p$ is above the hyperplane, and $|\overrightarrow{pp'}| = 0$ means $p$ is on the hyperplane.

According to (3) and (4),

If $|\overrightarrow{pp'}| < 0$:

$$\overrightarrow{pp'} = \begin{pmatrix} p_1 - f_1' \\ p_2 - f_2' \\ \vdots \\ p_m - f_m' \end{pmatrix} = \frac{-|\overrightarrow{pp'}|}{|\overrightarrow{n}|} \cdot \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

else $|\overrightarrow{pp'}| > 0$:

$$\overrightarrow{pp'} = \begin{pmatrix} p_1 - f_1' \\ p_2 - f_2' \\ \vdots \\ p_m - f_m' \end{pmatrix} = \frac{|\overrightarrow{pp'}|}{|\overrightarrow{n}|} \cdot \begin{pmatrix} -1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \tag{5}$$

. The mapping point $p'$ can be calculated according to (5). This way, all the normalized solutions can be linearly mapped onto the hyperplane.

Fig. 2 gives an illustrative example of one-to-one linear mapping, showing five normalized non-dominated solutions $a$, $b$, $p$, $c$ and $d$. Thus, the mapping point (namely, the pedal point) $p'$ of solution $p$ can be found by calculating the intersection between the hyperplane $f_1 + f_2 = 1$ and $pp'$ perpendicular to the hyperplane.

### B. Identification Stage

The identification stage in Algorithm 1 aims to group the solutions on the hyperplane into a number of candidate knee regions and identify knee points. Firstly, the density of each solution is calculated based on the harmonic mean distance between the solution and its neighbors in a predefined neighborhood. Then, the solutions are clustered according to the solution density, each group representing a candidate knee region. The convexity of each candidate knee groups will be determined and linear knee regions are discarded. The groups having a large curvature, which is indicated by a large difference between the radial coordinate value of the knee point and that of its neighbors, are kept. To reduce the influence of the inaccuracy in knee identification, a handful solutions near the identified knee points will be presented to the DM. The details are presented in Algorithm 2.

*1) Density estimation:* As illustrated in Fig. 2, the density of the solutions mapped onto the hyperplane $f_1 + f_2 = 1$ changes according to the curvature of the Pareto front on which the original solutions are located. More specifically, solutions in a knee region will have a smaller density on the hyperplane than that of the solutions located outside knee regions. For

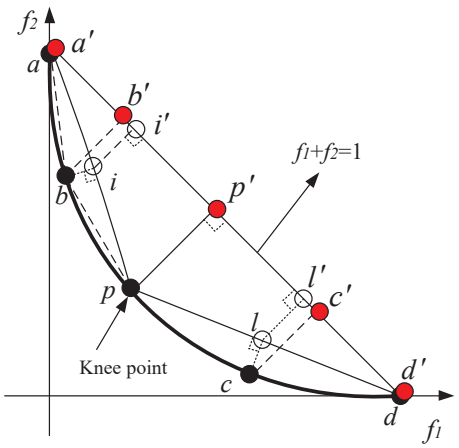



Fig. 2. An example of mapping the solutions onto a hyperplane for density estimation.

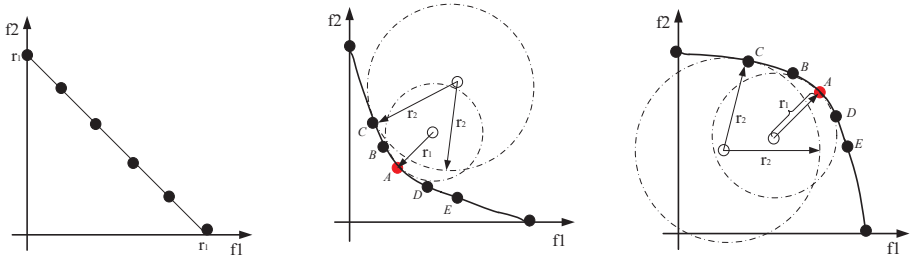
example in Fig. 2, $|\overrightarrow{ab}| = |\overrightarrow{bp}|$, and point $i$ is the pedal of solution $b$ on a local hyperplane $|ap|$. Points $b'$ and $i'$ are the pedal points of $b$ and $i$ on a local hyperplane $|a'd'|$. Thus, point $i$ and $i'$ are the middle point of $|a'p|$ and $|a'p'|$, namely, $|a'i'| = |i'p'|$. In other words, $|a'b'| < |b'p'|$, and likewise, $|c'd'| < |p'c'|$. Thus, it can be concluded that the density of $p'$ is smaller than its neighboring points.

Different from other methods for knee point identification, the method introduced in this paper transfers the curvature information into the difference in solution density on the hyperplane. Thus, knee points in convex or concave regions will have the smallest density. The next important question is how to calculate the density of each solution and how to find the solutions in the same knee region. Here we adopt a simple way to compute the density of the mapped solutions on hyperplane by measuring the closeness of a solution to its neighbors. The density of a solution $p$, $\rho$, can be calculated in the following steps:

1) Find $k$ nearest neighbors of each solution $p$.
2) Calculate the Euclidean distances between solution $p$ and its neighbors, denoted as $d_1, d_2, \cdots, d_k$, $k$ is the number of neighbors.
3) Compute the density of solution $p$, $\rho = 1/\overline{d}$, where $\overline{d} = \frac{k}{\frac{1}{d_1} + \cdots + \frac{1}{d_k}}$ is the harmonic mean distance.

Note that the harmonic mean distance is adopted because it favors shorter distances. In other words, the distance from $p$ to its closest neighbor plays a more important role in the mean distance. In Algorithm 2, Lines 5 to 9 calculate the density of each mapped solution on the hyperplane. $d_{1:\kappa} = minDis(x_i, \kappa)$ calculates the distance from $x_i$ to its nearest $\kappa$ neighbors, and $\overline{d}$ is the harmonic mean distance.

*2) Identification:* Lines 10 to 22 in Algorithm 2 identify knee regions on the basis of the estimated solution density. $\imath = arg\min_{x_i \in S}(\rho(x_i))$ means to find the index of the solution with the smallest density. In Line 11, $K = clustering(x_\imath, \varsigma)$ clusters the nearest $\varsigma$ solutions for solution $x_\imath$ and $\varsigma < \kappa$ aims




(a) Linear:$\sum_{i=1}^{m} f_i(x) = r$ (b) Convex: $\sum_{i=1}^{m}(r - f_i^2(x)) = r^2$ (c) Concave: $\sum_{i=1}^{m} f_i^2(x) = r$

Fig. 3. Determination of the convexity with the help of the radial coordinate value.

**Algorithm 2** : Knee identification by means of density estimation

**Input:** Transformed solution set: $S$, the size of neighborhood: $\kappa$,the size of the group: $\varsigma$, and $\varsigma < \kappa$, $\phi : 7 \pm 2$, threshold of difference: $diff$.

**Output:** Sorted solution set: $K^{'}$.

1: $K = \emptyset, B = \emptyset, K^{'} = \emptyset$.
2: $B = calc_b(S)$ %find the extreme points.%
3: $K^{'} = K^{'} \cup B$
4: $S = S/B$ %eliminate the extreme points.%
5: **for each** $x_i \in S$ **do**
6: $d_{1:\kappa} = minDis(x_i, \kappa)$ %calculate the distance from the solution to its nearest $\kappa$ solutions.%
7: $\overline{d} = k/(\frac{1}{d_1} + \cdots + \frac{1}{d_k})$ %harmonic distance.%
8: $\rho(x_i) = \frac{1}{\overline{d}}$ %the density of solution $x_i$.%
9: **end for**
10: **while** $S \neq \emptyset$ **do**
11: $\imath = arg\min_{x_i \in S}(\rho(x_i))$
12: $K = clustering(x_\imath, \varsigma)$
13: **if** $Distiguish(K, \varsigma) == true$ **then**
14: $K = Sorting(K, \phi, diff)$
15: $K^{'} = K^{'} \cup K$
16: $S = S/K$
17: $K = \emptyset$
18: **else**
19: $S = S/K$
20: $K = \emptyset$
21: **end if**
22: **end while**
23: $Output(K^{'})$

to reduce the influence of errors in the estimation. In Line 13, $Distinguish(K, \varsigma) == true$ verifies whether solution $x_\imath$ and its neighboring solutions form a convex or concave knee region.

In $Distinguish(K, \varsigma)$, the first step is to find out the boundary (extreme) points of cluster $K$ for solution $x_\imath$ and use them to construct a hyperplane. Then, the algorithm determines whether a candidate knee solution $x_\imath$ in region $K$ is below, above or on the hyperplane using (5), thereby preliminarily determining the convexity of the candidate knee regions. Candidate groups of solutions in a linear region will be discarded.

$$\begin{cases} \text{Linear:} & \sum_{i=1}^{m} f_i(x) = r \\ \text{Convex:} & \sum_{i=1}^{m}(r - f_i^2(x)) = r^2 \\ \text{Concave:} & \sum_{i=1}^{m} f_i^2(x) = r \end{cases} \tag{6}$$

To further verify whether the candidate knee regions identified with the help of density estimation are indeed knee regions, the radial coordinate value [30] described in (6) will be calculated for each solution. As shown in Fig. 3, the radial coordinate value of each solution varies and the larger the differences in the radial coordinate values, the stronger the curvature is. Thus, candidate knee regions having a large difference in the radial coordinate values of the solutions are confirmed to be knee regions. Let $r^o$, $r_1$, $\cdots$, $r_K$ denote the radial coordinate values of the candidate knee solutions and that of its $K$ neighbors, the curvature of a candidate knee region can be represented by $var$ calculated as follows:

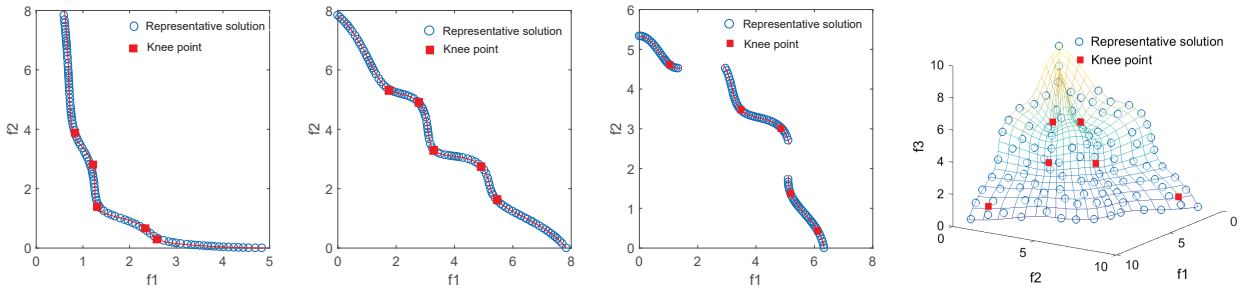
$$\begin{cases} var = |r^o - \overline{r}| \\ \text{where } \overline{r} = \frac{\sum_{i=1}^{K} r_i}{K} \end{cases} \tag{7}$$

A threshold $diff$ is defined to ensure that only candidate knee regions whose $var$ is larger than $diff$ are considered to be real knee regions. Finally, a small number of solutions in the neighborhood of the identified knee points will also be presented to the DM to reduce the influence of the estimation error.

## IV. Experimental studies

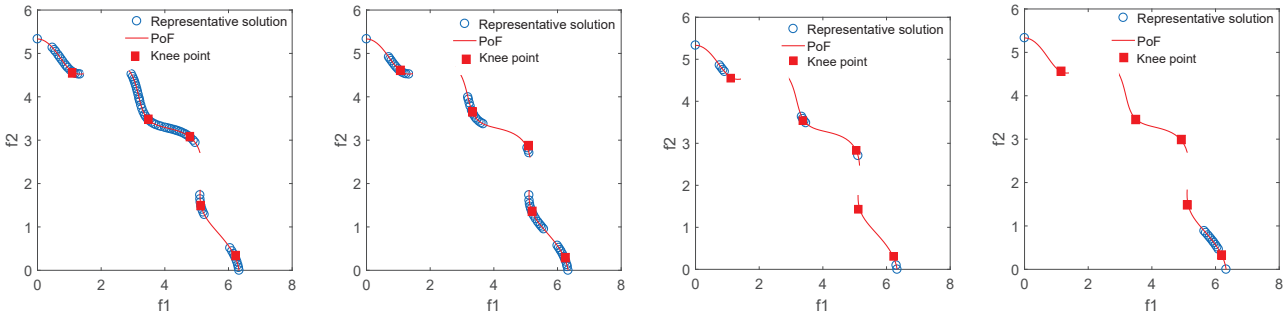
### A. Experimental settings

The basic evolutionary optimization framework is based on SPEA-II [2], and all compared algorithms are implemented in jMetal [31]. In the experiments, the distribution index is set to 10 for simulated binary crossover and 20 for polynomial mutation. The crossover probability and mutation probability are set to 0.99 and 0.1, respectively. The size of the population and archive are set to 100. SPEA-II is terminated when a maximum of 500 and 1000 generations are exhausted on 2-objective and 3-objective test problems, respectively. In identification of the knee points from the non-dominated solution sets obtained by the compared algorithms, $\kappa = 3 * \phi$ and $\varsigma = 2 * \phi$, where $\kappa$ is the size of the neighborhood for density estimation, and $\varsigma$ is



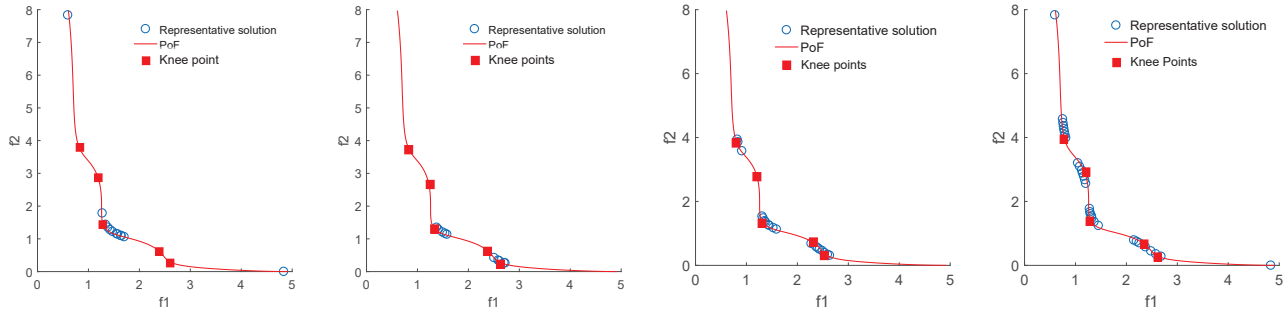

(a) DO2DK (b) DEB2DK (c) CKP (d) DEB3DK

Fig. 4. The solution set obtained by SPEA-II and the potential regions of interest on the PoF.




(a) 0.001 (b) 0.01 (c) 0.05 (d) 0.1

Fig. 5. The final obtained representative solutions in terms of different settings on parameter *diff* in KneeDEA.




(a) KneeDis (b) KneeEMU (c) KneeWD (d) KneeDEA

Fig. 6. The final representative solutions obtained by different methods on DO2DK.

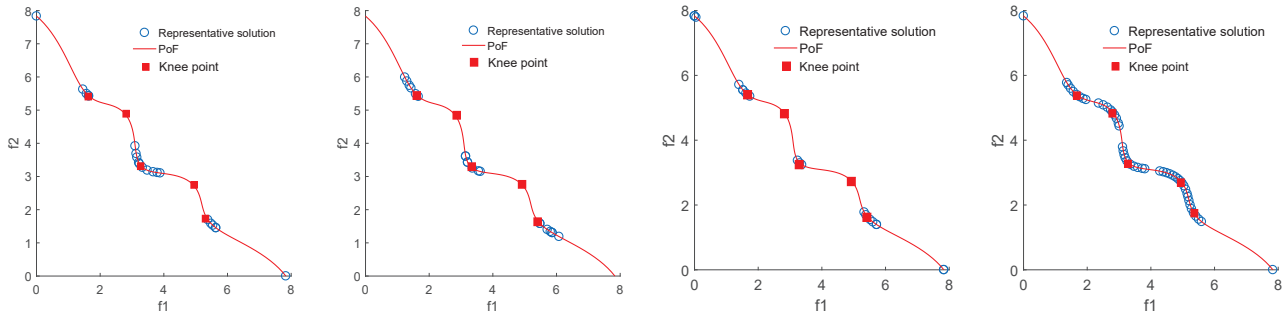
the size of the cluster, and $\phi = 7 \pm 2$ is the size of the solution group in the knee region.

To verify the effectiveness of the proposed identification method, three existing test problems, DO2DK, DEB2DK, and DEB3DK [26], are adopted. A new 2-objective test problem with a discrete and concave basic PoF, termed CKP, is proposed in this paper for comparing the knee identification methods.

The problem (CKP) is defined as follows:

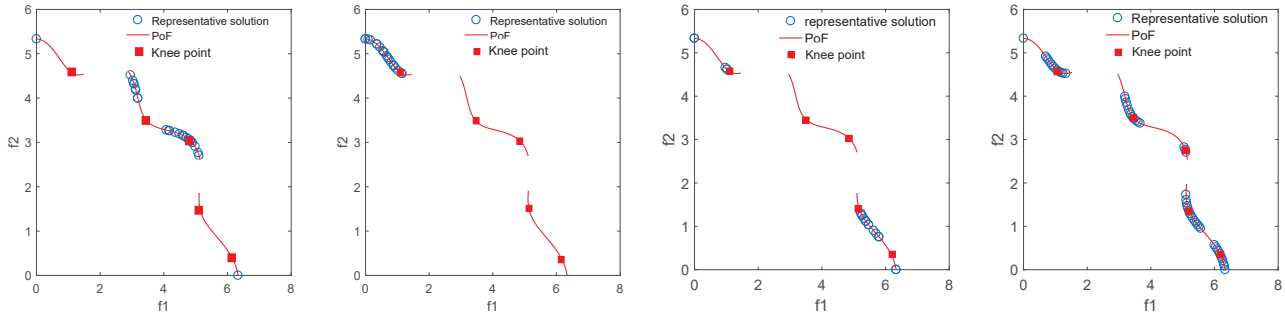
$$\begin{cases} \min: & f_1(x) = g(x)r(x_1)\sin(\pi x_1/2) \\ \min: & f_2(x) = g(x)r(x_1)\cos(\pi x_1/2) \\ & g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i \\ & r(x) = 5 + x_1^2 + \frac{\cos(2K\pi x_1/2)}{K} \\ & 0 \le x_i \le 1, i = 1, \ldots, n. \end{cases} \tag{8}$$

In the experiments, $K = 3, s = 1$ is set for DO2DK, and $K = 3$ for DEB2DK and CKP, and $K = 2$ for DEB3DK. The number of decision variables is set to 30 for all problems. The representative solution sets and corresponding knee regions on the PoFs are shown in Fig. 4, where the knee points are



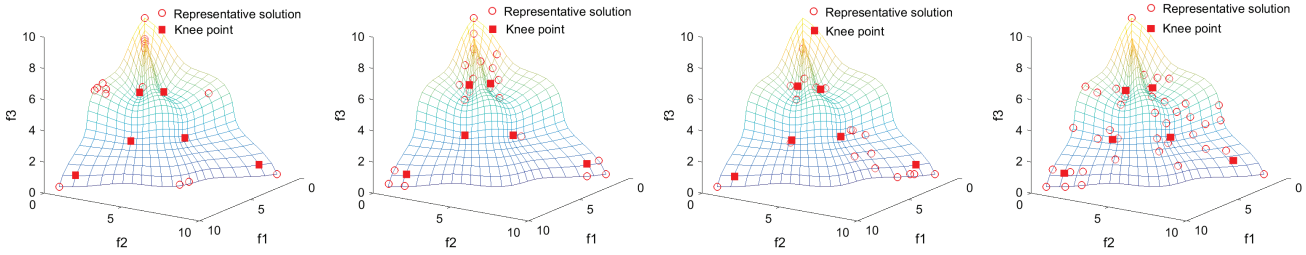

(a) KneeDis (b) KneeEMU (c) KneeWD (d) KneeDEA

Fig. 7. The final representative solutions obtained by different methods on DEB2DK.




(a) KneeDis (b) KneeEMU (c) KneeWD (d) KneeDEA

Fig. 8. The final representative solutions obtained by different methods on CKP.




(a) KneeDis (b) KneeEMU (c) KneeWD (d) KneeDEA

Fig. 9. The final representative solutions obtained by different methods on DEB3DK.

denoted using red squares. Note that DO2DK, DEB2DK, and CKP are bi-objective problems, and DEB3DK is a 3-objective problem. In Fig. 4, DO2DK has three convex knee points and two concave knee points. DEB2DK and CKP have similar characters as DO2DK except that the basic shapes of the PoFs are different. DEB3DK has six knee points, where four knees are located in convex regions and two located in concave regions. In the proposed method, the threshold value *diff* is set to $0.01$ for CKP and to $0.1$ for the others.

Three knee point identification methods are taken from the literature for comparison. The identification method [25], KneeDis for short, is based on the distances to a hyperplane (constructed using the extreme points) to identify knee regions. In [23], the expected marginal utility is adopted to identify knees, which we term KneeEMU. Finally, the method in [29] is based on a different working definition of the knee points, which is termed KneeWD for convenience. $\triangle = 0.35$ is set for KneeDis, and $\delta = 0.2$ for KneeWD. The method proposed in this work is named KneeDEA for short.

### *B. Parameter sensitivity analysis*

In the proposed identification method, *diff* is a key parameter which can control the number of knee regions to be identified. Fig. 5 shows four groups of knee solutions obtained when *diff* is set to 0.001, 0.01, 0.005 and 0.1, respectively. From these results, we can see that a smaller *diff* will lead

to more knee groups while a larger $diff$ will result in fewer groups of knee points. Therefore, if the DM is interested in exploring more groups of possible knee regions, a smaller $diff$ should be used.

However, it is non-trivial to set a proper value for $diff$ when optimizing a problem with little a priori knowledge. In practice, the user can start with a small value for $diff$, say 0.001. If the number of resulting groups is too large, gradually increase the value until desired number of groups are obtained. Adaptive tuning of the $diff$ value will be our future work.

### C. Comparative experiments

The experiments here aim to examine the effectiveness of the proposed method in identifying knees in both convex and concave regions.

The results on DO2DK obtained by the four algorithms under comparison are shown in Fig. 6. From the figure, we can see that all methods are able to identify the convex knee in the middle of the PoF. However, KneeDis has missed all other knee regions, mainly because the distance from the knee point in the missed knee regions to the hyperplane is smaller than the distance from some of its neighboring points to the hyperplane. We also note that both KneeEMU and KneeWD can identify all convex knee regions while missing the concave ones. By contrast, KneeEDA has successfully identified all concave and convex knees. One issue with KneeEDA is that it is not able to distinguish two very close knee regions near the right end of the PoF, as shown in Fig. 6 (d).

Fig. 7 shows the knee regions of DEB2DK detected by the compared algorithms. All methods can identify the convex knees, and all but KneeEDA fail to detect the concave knee regions. It is because the density of the solution is obvious and the difference between the potential knee and its neighbors is distinguishable. Again, KneeEDA is not able to distinguish two very close knee regions.

Since CKP has a discrete and concave basic PoF, it poses slightly different difficulties to the compared knee detection algorithms. From Fig. 8, we note that KneeDis can find knees that are far from the hyperplane. KneeEMU largely fails to work properly because the boundary points have higher priority in terms of the linear utility function. KneeWD and KneeEMU experience similar problems. Among the compared algorithms, only KneeEDA is able to identify all knee regions.

Finally, the results on the DEB3DK are provided in Fig. 9. It can be seen that KneeDis completely fails to work on DEB3DK since none of the knee regions have been correctly detected. KneeEMU is able to identify the concave knees since the concave knees on the PoF of DEB3DK are closer to the boundary points, which have a high priority in identification. KneeWD exhibits better performance than KneeDis and KneeEMU in identifying convex knees on this test problem, yet it is not able to detect the concave knees. KneeEDA appears to be able to identify both convex and concave knee regions, however, it is not able to clearly distinguish different knee regions that are close to each other.

From the results presented above, we can conclude that the proposed method is able to identify both convex and concave knee regions. Moreover, its performance is less sensitive to the convexity or continuity of the basic shape of the PoF. One issue with the proposed method is that it may not be able to distinguish knee regions that are close to each other.

## V. Conclusion

Most research on evolutionary multi-objective optimization has focused on finding well distributed and highly converged non-dominated solutions but much less has been devoted to studying methods for selecting a small set of solutions to be presented to the DM. Without specific preference information provided by the DM, knee points are typically considered to be the most preferred solutions.

Although a few algorithms have been proposed to detect knee points, most of them are not able to detect knees in concave regions. The performance of some existing algorithms is also sensitive to the shape of the basic Pareto front. To resolve these issues, this paper proposes a novel method based on solution density. After mapping the normalized non-dominated solutions to a constructed hyperplane, the algorithm identifies candidate knee regions based on the estimated density of the projected solutions. Then the curvature of the detected knee regions will be calculated to determine the convexity of the candidate knee regions and finally, a threshold on the curvature is used to determine the knee solutions to be presented to the DM.

Empirical studies are conducted on four test problems and the results confirm that the proposed method is able to identify both convex and concave knees. The proposed method is also able to detect knee regions that are far from the middle of the Pareto front and is less sensitive to the convexity of the basic shape of the PoF.

Note that an implicit assumption of the proposed knee identification algorithm is that a well distributed and accurate solution set has been achieved. If the obtained solution set is not well distributed or not accurate, the algorithm may be misled, since the density of the projected solutions may not reflect the curvature of the true Pareto front.

Apart from the above assumption, the proposed method is also subject to the following weaknesses. First, it cannot separate close knee regions. Second, as discussed before, it is not straightforward to define a proper value for $diff$ when little knowledge is available for a problem to be solved. Thus, one future work is to investigate the possibility of adaptively tuning the threshold rather than pre-specifying it by the DM. Finally, the proposed algorithm is limited to bi- or three-objective optimization problems, as solution density only will not be sufficient in a high-dimensional objective space. In this case, it will be more realistic to identify knee regions interactively during the optimization process.

## Acknowledgment

This work was supported in part by the Honda Research Institute Europe GmbH.

## REFERENCES


[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[2] E. Ziztler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization," *in Proc. EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control With Applications to Industrial Problems, K. Giannakoglou, D. Tsahalis, J. Periaux, P. Papailou, and T. Fogarty, Eds., Athens, Greece, Sept*, 2001.

[3] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, 2002.

[4] G. Yu, R. Shen, J. Zheng, M. Li, J. Zou, and Y. Liu, "Binary search based boundary elimination selection in many-objective evolutionary optimization," *Applied Soft Computing*, vol. 60, pp. 689–705, 2017.

[5] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 832–842.

[6] 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, 2007.

[7] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.

[8] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[9] E. J. Hughes, "Multiple single objective Pareto sampling," in *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, vol. 4. IEEE, 2003, pp. 2678–2684.

[10] 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. Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[11] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.

[12] A. Sinha, P. Korhonen, J. Wallenius, and K. Deb, "An interactive evolutionary multi-objective optimization method based on polyhedral cones," in *International Conference on Learning and Intelligent Optimization*. Springer, 2010, pp. 318–332.

[13] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 474–494, 2013.

[14] G. Yu, J. Zheng, R. Shen, and M. Li, "Decomposing the user-preference in multiobjective optimization," *Soft Computing*, vol. 20, no. 10, pp. 4005–4021, 2016.

[15] J. Zheng, G. Yu, Q. Zhu, X. Li, and J. Zou, "On decomposition methods in interactive user-preference based optimization," *Applied Soft Computing*, vol. 61, pp. 806–818, 2017.

[16] P. Agarwal, M. Sahai, V. Mishra, M. Bag, and V. Singh, "A review of multi-criteria decision making techniques for supplier evaluation and selection," *International Journal of Industrial Engineering Computations*, vol. 2, no. 4, pp. 801–810, 2011.

[17] L. Rachmawati and D. Srinivasan, "Preference incorporation in multiobjective evolutionary algorithms: A survey," in *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2006, pp. 962–968.

[18] H. Wang, M. Olhofer, and Y. Jin, "A mini-review on preference modeling and articulation in multi-objective optimization: current status and challenges," *Complex & Intelligent Systems*, pp. 1–13, 2017.

[19] J. Dyer, "Multiple criteria decision analysis: state of the art surveys," *International Series in Operations Research and Management Science*, vol. 78, no. 4, pp. 265–292, 2005.

[20] M. A. Braun, P. K. Shukla, and H. Schmeck, "Preference ranking schemes in multi-objective evolutionary algorithms," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2011, pp. 226–240.

[21] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015.

[22] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Engineering Optimization*, vol. 43, no. 11, pp. 1175–1204, 2011.

[23] K. Bhattacharjee, H. Singh, M. Ryan, and T. Ray, "Bridging the gap: Many-objective optimization and informed decision-making," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 5, pp. 813–820, 2017.

[24] I. Das, "On characterizing the "knee" of the Pareto curve based on normal-boundary intersection," *Structural and Multidisciplinary Optimization*, vol. 18, no. 2, pp. 107–115, 1999.

[25] O. Schütze, M. Laumanns, and C. A. C. Coello, "Approximating the knee of an MOP with stochastic search algorithms." in *PPSN*. Springer, 2008, pp. 795–804.

[26] J. Branke, K. Deb, H. Dierolf, and M. Osswald, "Finding knees in multiobjective optimization," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 722–731.

[27] M. Braun, P. Shukla, and H. Schmeck, "Angle-based preference models in multi-objective optimization," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2017, pp. 88–102.

[28] S. Bechikh, L. B. Said, and K. Ghédira, "Searching for knee regions of the Pareto front using mobile reference points," *Soft Computing*, vol. 15, no. 9, pp. 1807–1823, 2011.

[29] L. Rachmawati and D. Srinivasan, "Multiobjective evolutionary algorithm with controllable focus on the knees of the Pareto front," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 810–824, 2009.

[30] Z. He and G. G. Yen, "Visualization and performance metric in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 386–402, 2016.

[31] J. J. Durillo and A. J. Nebro, "jMetal: A java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, no. 10, pp. 760–771, 2011.