On the Normalization in Evolutionary Multi-Modal Multi-Objective Optimization

Yiping Liu[1], Hisao Ishibuchi[2], Gary G. Yen[3], Yusuke Nojima[1], Naoki Masuyama[1], and Yuyan Han[4]
[1]Department of Computer Science and Intelligent Systems,
Osaka Prefecture University, Sakai, Osaka 599-8531, Japan
[2]Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation,
Department of Computer Science and Engineering,
Southern University of Science and Technology, Shenzhen 518055, China
[3]School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA
[4]School of Computer Science, Liaocheng University, Liaocheng, 252059, China
Email: yiping0liu@gmail.com, hisao@sustech.edu.cn, gyen@okstate.edu,
nojima@cs.osakafu-u.ac.jp, masuyama@cs.osakafu-u.ac.jp, hanyuyan@lcu-cs.com

***Abstract*—Multi-modal multi-objective optimization problems may have different Pareto optimal solutions with the same objective vector. A number of evolutionary multi-modal multi-objective algorithms have been developed to solve these problems. They aim to search for a Pareto optimal solution set with good diversity in both the objective and decision spaces. Although the normalization in both the objective and decision spaces is very important for these algorithms, there are few studies on this topic. In this paper, we investigate the effect of four normalization methods on two evolutionary multi-modal multi-objective algorithms. Six distance minimization problems are chosen as test problems. The experimental results show that the effect of normalization in evolutionary multi-modal multi-objective optimization is algorithm- and problem-dependent.**

***Index Terms*—multi-modal multi-objective evolutionary optimization, normalization, objective space, decision space**

I. Introduction

An optimization problem, such as job shop scheduling [1] and financial portfolio management [2], may have multiple objectives which conflict with each other. Such a problem is regarded as a multi-objective optimization problem (MOP). There is no single optimal solution for the problem. Instead, it has a set of Pareto optimal solutions, i.e., the Pareto optimal solution set (PS). The image of PS in the objective space is called the Pareto front (PF). Without loss of generality, an MOP with box constraints can be formulated as follows:

$$\begin{aligned} &\min \boldsymbol{f}(\boldsymbol{x}) = \min(f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})), \\ &\text{s.t. } \boldsymbol{x} \in S, \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is an n-dimensional decision vector in the feasible space S, $f_m(\boldsymbol{x})$ is the m-th objective to be minimized ($m = 1, ..., M$), and M is the number of objectives. $S = \{\boldsymbol{x} \in \boldsymbol{R}^n : x_i^{\text{lower}} \le x_i \le x_i^{\text{upper}}, i = 1, \ldots, n\}$ where x_i^{upper} and x_i^{lower} are the lower and upper bounds of x_i, respectively.

An MOP may have multiple Pareto optimal solutions corresponding to the same objective vector, which are called equivalent Pareto optimal solutions. In such a case, the MOP is termed as a multi-modal multi-objective optimization problem (MMOP). In recent years, a number of evolutionary multi-modal multi-objective algorithms (EMMAs) have been developed to solve MMOPs. The general goal of EMMAs is to search for a set of Pareto optimal solutions with good diversity both in the objective and decision spaces. Based on the diversity evaluation mechanisms, existing EMMAs can be roughly classified into two groups. One evaluates the diversity in the objective and decision spaces simultaneously, while the other separately.

The first group includes Omni-optimizer [3], Niching-covariance matrix adaptation (Niching-CMA) [4], double-niched evolutionary algorithm (DNEA) [5], [6], multi-objective particle swarm optimization algorithm using ring topology and special crowding distance (MO_Ring_PSO_SCD) [7], and multi-modal multi-objective differential evolution optimization algorithm (MMODE) [8]. In the environmental selection, these algorithms usually use the Pareto rank (i.e., the front number according to the non-dominated sorting [9]) as the primary selection criterion. For solutions with the same Pareto rank, they use the objective and decision space density values to select solutions with good diversity. One benefit for these EMMAs is that there exist a larger number of state-of-the-art density evaluation methods to choose. For example, DNEA uses two niche-based sharing functions [10] to estimate the objective and decision space density values, respectively, and then sum them together. In Omni-optimizer, MO_Ring_PSO_SCD, and MMODE, the objective and decision space density values are estimated by the crowding distance [9]. However, one open issue of the EMMAs in this group is how to balance the

This work was supported in part by the National Natural Science Foundation of China under Grant 61876075 and Grant 61803192, in part by the Guangdong Provincial Key Laboratory under Grant 2020B121201001, in part by the Program for Guangdong Introducing Innovative and Enterpreneurial Teams under Grant 2017ZT07X386, in part by the Shenzhen Science and Technology Program under Grant KQTD2016112514355531, in part by the Program for University Key Laboratory of Guangdong Province under Grant 2017KSYS008, and in part by Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP19K20358.

Corresponding Author: Hisao Ishibuchi

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

diversities in the objective and decision spaces.

Multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies (TriMOEA-TA&R) [11] and multi-objective evolutionary algorithm based on decomposition with addition and deletion operators (MOEA/D-AD) [12] are typical algorithms in the other group. They use a set of uniformly distributed reference vectors to guarantee diversity in the objective space. Then, for each reference vector, they try to search for diverse Pareto optimal solutions in the decision space. These EMMAs may have a strength in handling the inconsistency between the diversities in the objective and decision spaces. However, it is usually difficult to specify the reference vectors for a real-world MMOP.

One exception is a recently proposed evolutionary algorithm using a convergence-penalized density method (CPDEA) [13]. CPDEA uses the convergence-penalized density method as a selection criterion in the environmental selection, which is different from those in the aforementioned two groups. It can handle imbalanced MMOPs where the difficulties of finding equivalent Pareto optimal solutions are different. CPDEA has an extra archive which uses a similar selection criterion with those in the first group.

Although a number of EMMAs have been proposed, there are few discussions on the normalization in evolutionary multi-modal multi-objective optimization, which is quite important for EMMAs, especially those in the first group. If the scales of the PF and PS are very different, the objective and decision space density values of a solution are usually very different. Then, either the objective space density value or the decision space density value plays a leading role in the environmental selection. This may result in losing diversity either in the decision or objective space. One way to solve this issue is to normalize the objective and decision values of a solution into similar scales based on the true PF and PS, respectively. Since the true PF and PS are usually unknown, we can use some information before or during the evolution. There have been several studies on normalizing the objective values [14]–[16]. One of the most popular ways is to normalize the objective values of a solution based on the objective values of the non-dominated solutions during the evolution. However, normalizing the decision values of a solution based on the decision values of the non-dominated solutions during the evolution may cause some issues. For example, if the PS is a line segment parallel to one coordinate axis, using such a normalization method will result in a convergence issue. In such a case, it may be better to normalize the objective and decision space density values into similar scales, rather than the objective and decision values.

In this paper, we investigate the effect of four normalization methods on EMMAs in the first group. In the first three methods, the objective values of a solution are normalized based on the objective values of the non-dominated solutions in the current population. The decision values of a solution are normalized based on the range of the feasible space, the decision values of all the solutions in the current population, and the decision values of the non-dominated solutions in the current population in the first, second, and third methods, respectively. In the fourth method, the objective and decision space density values of solutions are normalized based on the mean distances among solutions in the objective and decision spaces, respectively.

The remainder of this paper is organized as follows. In Section II, we describe the EMMAs examined in our experiments. The four normalization methods are explained in detail in Section III. Section IV shows the experimental design and results. Section V concludes the paper and provides future research directions.

II. Evolutionary Multi-Modal Multi-Objective Algorithm

The framework of a conventional EMMA is shown as Algorithm 1. In Algorithm 1, a population P of size N is randomly initialized (line 1). The stopping criterion is usually a predefined number of evaluated solutions. While the stopping criterion has not been met (line 2), an offspring population Q is generated by evolutionary reproduction operators (line 3). In this study, we use simulated binary crossover and polynomial mutation to generate Q. Then, P in the next generation is selected from $P \cup Q$ by the environmental selection operator (line 4). At the end of evolution, P is returned (line 6).

Algorithm 1 Framework of EMMA

Require: P (Population), N (Population Size)
1: P = *Initialization* (P);
2: **while** the stopping criterion is not met **do**
3: Q = *Reproduction* (P);
4: P = *Environmental_Selection* ($P \cup Q$);
5: **end while**
6: **return** P

In this study, we use the Pareto rank as the primary selection criterion in the environmental selection. Since the crowding distance is one of the most popular secondary selection criterion, we choose a special crowding distance (SCD) [7] as the secondary selection criterion. SCD is modified from the density evaluation method in Omni-optimizer. In SCD, the crowding distance of a boundary solution in the objective space is no longer infinite. This makes SCD more appropriate for multi-modal multi-objective optimization than its original version. After calculating the crowding distances in both the objective and decision spaces, the SCD value of a solution is specified as follows:
If $\mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x}) > \mathrm{CD}^{\mathrm{obj}}_{\mathrm{mean}}$ or $\mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x}) > \mathrm{CD}^{\mathrm{dec}}_{\mathrm{mean}}$,

$$\mathrm{SCD}(\boldsymbol{x}) = \max(\mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x}), \mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x})), \tag{2}$$

where $\mathrm{CD}^{\mathrm{obj}}_{\mathrm{mean}}$ and $\mathrm{CD}^{\mathrm{dec}}_{\mathrm{mean}}$ are the mean crowding distances in the objective and decision spaces, respectively;
otherwise,

$$\mathrm{SCD}(\boldsymbol{x}) = \min(\mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x}), \mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x})). \tag{3}$$

In addition, we use a variant of SCD in this study. That is,

$$\mathrm{SCD}'(\boldsymbol{x}) = \mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x}) + \mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x}). \tag{4}$$

Hereafter, the EMMAs using SCD and SCD′ are denoted as EMMA1 and EMMA2, respectively. We will investigate the effect of normalization on their performance through computational experiments.

III. Four Normalization Methods

Four normalization methods to be investigated are described in detail in the following.

In the first method, the objective values of each solution are normalized by

$$f_m(\boldsymbol{x}) = (f_m(\boldsymbol{x}) - f_m^{\min})/(f_m^{\max} - f_m^{\min}), \quad m = 1, \ldots, M, \tag{5}$$

where $f_m^{\min}$ and $f_m^{\max}$ are the minimum and maximum values of the m-th objective of the non-dominated solutions in the current population, respectively.

The decision values of each solution are normalized as follows.

$$x_i = (x_i - x_i^{\text{lower}})/(x_i^{\text{upper}} - x_i^{\text{lower}}), \quad i = 1, \ldots, n, \tag{6}$$

where x_i^{lower} and x_i^{upper} are the lower and upper bounds of x_i in the box constraint, respectively.

In the second method, the normalization in the objective space is the same with that in the first method. The decision values of each solution are normalized as follows.

$$x_i = (x_i - x_i^{\text{Pmin}})/(x_i^{\text{Pmax}} - x_i^{\text{Pmin}}), \quad i = 1, \ldots, n, \tag{7}$$

where x_i^{Pmin} and x_i^{Pmax} are the minimum and maximum values of the i-th decision variable of all the solutions in the current population, respectively.

In the third method, the objective values of each solution are also normalized using (5). The decision values of each solution are normalized by

$$x_i = (x_i - x_i^{\text{Nmin}})/(x_i^{\text{Nmax}} - x_i^{\text{Nmin}}), \quad i = 1, \ldots, n, \tag{8}$$

where x_i^{Nmin} and x_i^{Nmax} are the minimum and maximum values of the i-th decision variable of the non-dominated solutions in the current population, respectively.

In the fourth method, the objective and decision values of each solution are normalized using (5) and (6), respectively. Then, the objective and decision space density values (i.e., the crowding distances $\mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x})$ and $\mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x})$ in this study) of each solution are normalized as follows.

$$\begin{aligned} \mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x}) &= \mathrm{CD}^{\mathrm{obj}}(\boldsymbol{x})/d_{\text{mean}}^{\text{obj}}, \\ \mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x}) &= \mathrm{CD}^{\mathrm{dec}}(\boldsymbol{x})/d_{\text{mean}}^{\text{dec}}, \end{aligned} \tag{9}$$

where $d_{\text{mean}}^{\text{obj}}$ ($d_{\text{mean}}^{\text{dec}}$) is the mean distance between each non-dominated solution and its closest non-dominated solution in the objective (decision) space. That is,

$$\begin{aligned} d_{\text{mean}}^{\text{obj}} &= \textstyle\sum_{k=1}^{K} d_k^{\text{obj}}/K, \\ d_{\text{mean}}^{\text{dec}} &= \textstyle\sum_{k=1}^{K} d_k^{\text{dec}}/K, \end{aligned} \tag{10}$$

where K is the number of the non-dominated solutions in the current population, and d_k^{obj} (d_k^{dec}) is the distance between the k-th non-dominated solution and its closest non-dominated solution in the objective (decision) space.

In the above four methods, if there is only a single non-dominated solution in the population, the maximum and minimum objective (decision) values of the non-dominated solution are replaced by the maximum and minimum objective (decision) values of all the solutions in the population. Hereafter, EMMA1 or EMMA2 with the X-th normalization method is denoted as EMMA1-NX or EMMA2-NX.

It is worth noting that there are many other normalization methods. For example, we can perform the normalization based on the ranges of all solutions ever found during the evolution. For another example, we can use a scaling method based on sigmoid functions [17]. Due to space limitation, we will investigate the effect of those normalization methods in the future.

IV. Experiments

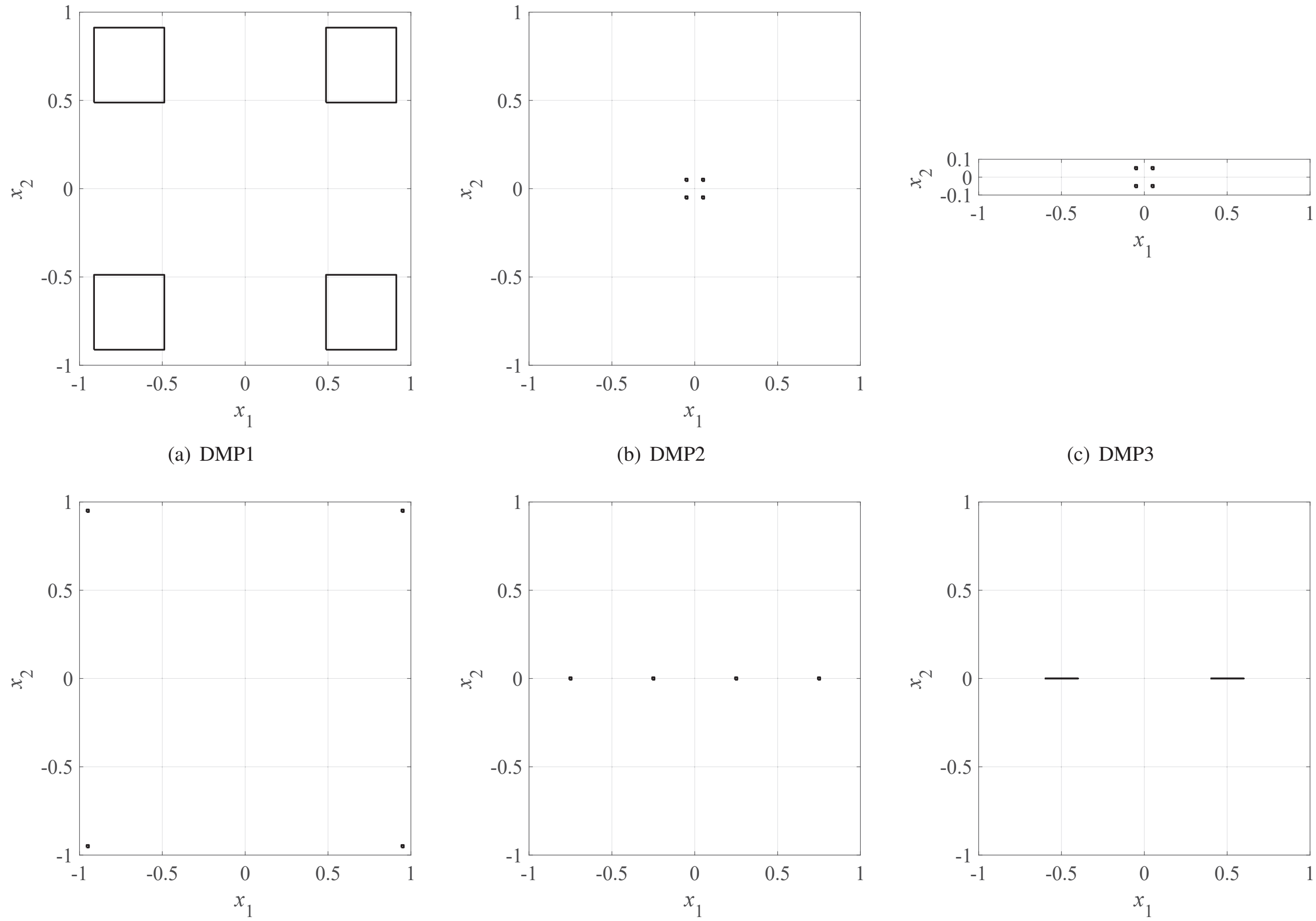
A. Test Problems

We choose six instances of distance minimization problems (DMPs) [18], [19] in our experiments. These instances are denoted as DMP1-6. For DMP1-5, the objectives are to minimize the distances to the vertexes of four rectangles in the two-dimensional decision space. That is, they have four objectives and four equivalent Pareto optimal regions. DMP6 has two objectives, which are to minimize the distances to the endpoints of two line segments in the two-dimensional decision space. The m-th ($m = 1, \ldots, M$) objective function of a DMP is defined as follows.

$$\begin{aligned} &f_m(\boldsymbol{x}) = \min\nolimits_{p=1,\ldots,P} \{d(\boldsymbol{X}_{p,m}, \boldsymbol{x})\}, \\ &\text{s.t. } x_i^{\text{lower}} \le x_i \le x_i^{\text{upper}}, i = 1, 2 \end{aligned} \tag{11}$$

where $\boldsymbol{X}_{p,m}$ is the m-th vertex (or endpoint) of the p-th polygon (or line segment), and $d(\boldsymbol{X}_{p,m}, \boldsymbol{x})$ is the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{X}_{p,m}$ in the decision space.

In each test problem, the size and the shape of each polygon (or line segment) are the same. Thus the polygons (or line segments) are equivalent Pareto optimal regions. That is, each region corresponds to the entire PF. The equivalent Pareto optimal regions of DMP1-6 are shown in Fig. 1. The sizes of the feasible region, the PS, and an equivalent Pareto optimal region of each DMP are listed in Table I.



(a) DMP1 (b) DMP2 (c) DMP3

(d) DMP4 (e) DMP5 (f) DMP6

Fig. 1. The equivalent Pareto optimal regions of DMP1 to DMP6. In DMP1 to DMP5, they are four rectangles. In DMP6, they are two line segments.

TABLE I
THE SIZES OF THE FEASIBLE REGION, THE PS, AND AN EQUIVALENT PARETO OPTIMAL SUBSET OF EACH DMP.

Size	Feasible Region $x_1 \times x_2$	PS $x_1 \times x_2$	Equivalent Pareto Optimal Region $x_1 \times x_2$
DMP1	2 × 2	1.91 × 1.91	0.42 × 0.42
DMP2	2 × 2	0.114 × 0.114	0.014 × 0.014
DMP3	2 × 0.2	0.114 × 0.114	0.014 × 0.014
DMP4	2 × 2	1.914 × 1.914	0.014 × 0.014
DMP5	2 × 2	1.014 × 0.014	0.014 × 0.014
DMP6	2 × 2	1.2 × 0	0.2 × 0

Based on Fig. 1 and Table I, the characteristics of DMP1-DMP6 are summarized as follows.

- DMP1 is a normalized problem. The differences among the sizes of the feasible region, the PS, and an equivalent Pareto optimal region in DMP1 are not large. DMP1 is expected to be easy for all the normalization methods.
- In DMP2, the size of an equivalent Pareto optimal region is smaller than that of the PS. Both of them are much smaller than the size of the feasible region.
- DMP3 is similar to DMP2. The difference is that, in DMP3, the feasible range of x_1 is much larger than that of x_2.
- In DMP4, the sizes of the feasible region and the PS are similar. However, they are much larger than the size of an equivalent Pareto optimal region.
- In DMP5, the PS size of x_1 is much larger than that of x_2.
- In DMP6, the PS size of x_2 is 0. Normalizing the x_2 value of each solution may result in critical issues.

B. Parameter Settings and Performance Indicator

Both EMMA1 and EMMA2 with four normalization methods are applied to each test problem 40 times. Such a number of run times is enough for the hypothesis test of the results. For each run, the following specifications are used:

- Population size: 200
- Population initialization: random points in the entire decision space

- Termination condition: 40,000 evaluations of solutions (to make sure that both EMMAs converge on every test problem)
- Crossover operator: SBX with $\eta_c = 20$
- Crossover probability: 1.0
- Mutation operator: Polynomial mutation with $\eta_m = 20$
- Mutation probability: 0.5

We use IGDX [20] to compare different methods. IGDX is a variant of Inverted Generational Distance (IGD). IGD comprehensively quantifies both convergence and diversity of an approximate solution set in the objective space, while IGDX quantifies that in the decision space. That is, IGDX measures the distance between the approximate solution set and a reference solution set, which is typically uniformly distributed on the true PS. The smaller value of IGDX, the better performance of the approximate solution set. For every test problem in this study, since the diversities in the objective and decision spaces are consistent, a small IGDX value usually indicates a small IGDX value. Thus we only use IGDX in the experiments. The size of reference solution set is over 1,000 for each test problem in this study.

C. Results and Discussions

1) Results of EMMA1: In Table II, we show the mean values and standard deviations of IGDX and the corresponding performance scores [21] of EMMA1 with the four normalization methods on DMP1-6. For each test problem, the performance score of a method is the number of the other methods which perform significantly worse than it according to IGDX. Here, the Wilcoxon's rank sum test is employed to determine whether a method shows a statistically significant difference from another, and the null hypothesis is rejected at a significant level of 0.05.

We can see from Table II that, surprisingly, the results of EMMA1 with the four normalization methods do not show any significant difference on any test problems. One possible reason is that the performance of EMMA1 is not stable due to the special crowding distance, which makes the standard deviations of IGDX quite large. Then, the effect of normalization on the IGDX values is not obvious. Actually, there are large differences in the mean IGDX values among the four methods (e.g., see the results on DMP4-6 in Table II).

2) Results of EMMA2: In the same manner as in Table II, we show the results of EMMA2 in Table III. From the performance scores in Table III, we can see that the performance of EMMA2 is sensitive to the choice of a normalization method. To visually investigate the behavior of EMMA2, in Figs. 2-5, we show the solutions obtained by EMMA2 with a different normalization method in a single run on DMP1-3, and DMP6, respectively. For each normalization method, we select the single run with the closest IGDX value to to the mean IGDX value in Table III. Since the obtained solutions of DMP4 and DMP5 are difficult to clearly visualize, in Table IV, we show the number of runs where EMMA2 found at least one solution in every equivalent Pareto optimal region on DMP4 and DMP5.

Based on Tables III-IV and Figs. 2-5, we have the following observations.

DMP1: For DMP1, EMMA2-N4 is slightly better than the others. However, the differences of the IGDX mean values among these methods are not large. The effect of normalization is not obvious on this test problem.

DMP2: EMMA2-N4 performs the best on DMP2, followed by EMMA2-N3. EMMA2-N1 and EMMA2-N2 perform poorly due to inappropriate normalization. In EMMA2-N1, the crowding distances in the decision space are usually much smaller than those in the objective space. In EMMA2-N2, the current population often provides misleading information for normalizing the decision values.

DMP3: DMP3 is modified from DMP2, where the feasible range of x_1 is much larger than that of x_2 in DMP3. Due to this characteristic, the performance scores of EMMA2-N1 and EMMA2-N4 on DMP3 become worse than those on DMP2. Although both EMMA2-N4 and EMMA2-N3 have the highest performance scores, the mean IGDX value of EMMA2-N4 is larger than that of EMMA2-N3. EMMA2-N1 is significantly worse than the others.

DMP4: For DMP4, EMMA2-N4 significantly outperforms the others. We can see from Table IV that, in most runs, EMMA2-N1, EMMA2-N2, and EMMA2-N3 cannot find any solution in some equivalent Pareto optimal regions. This is because all the first three normalization methods make the crowding distances in the decision space much smaller than those in the objective space, since the range of solutions in the decision space is much larger than that in the objective space.

DMP5: EMMA2-N4 is also the best on DMP5. All the first three normalization methods cannot work properly, especially the third one. Since the PS range of x_1 is much larger than that of x_2, normalizing the decision values of a solution based on the non-dominated solutions is quite misleading. As a result, EMMA2-N3 is more likely to lose solutions in some equivalent Pareto optimal regions (see Table IV), and thus obtained the largest mean value of IGDX.

DMP6: EMMA2-N4 achieved the highest performance score on DMP6, followed by EMMA2-N1, EMMA2-N2, and EMMA2-N3. The third normalization method encounters great difficulty since the PS range of x_2 is zero in DMP6. We can see from Fig. 5 that most solutions obtained by EMMA2-N1, EMMA2-N2, and EMMA2-N4 focus on the two line segments. However, solutions obtained by EMMA2-N3 widely spread away from the two line segments.

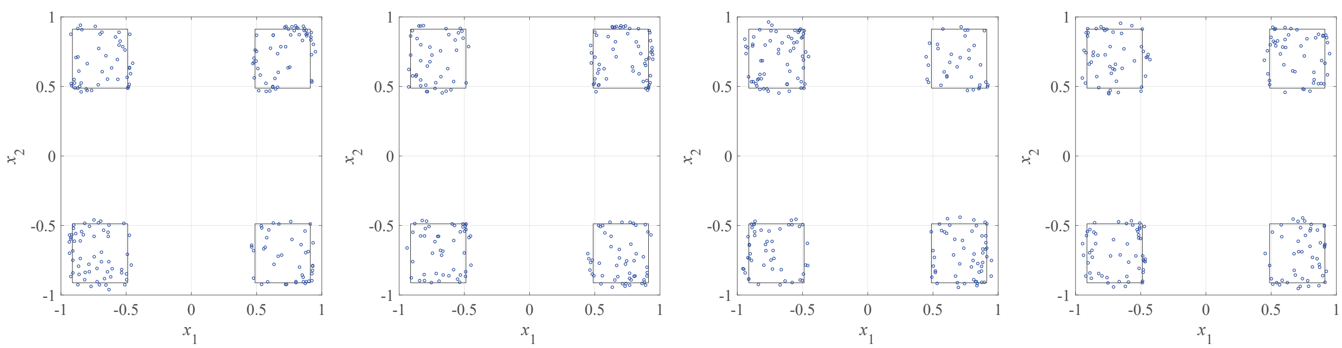
3) Comparison between EMMA1 and EMMA2: In Table V, we show the performance scores of all the eight methods on DMP1 to DMP6. We can see from Table V that EMMA2 is generally better than EMMA1 on these test problems except for DMP4. Although EMMA2-N4 apparently outperforms EMMA1 on DMP4, EMMA2 with the first three normalization methods are significantly worse than EMMA1. This indicates that the performance of EMMA2 may dramatically deteriorate when it uses incorrect normalization. This shows that EMMA2 is very sensitive to the choice of a normalization method on some test problems.

TABLE II
RESULTS OF EMMA1

IGDX	EMMA1-N1			EMMA1-N2			EMMA1-N3			EMMA1-N4		
	Mean	Deviation	Score	Mean	Deviation	Score	Mean	Deviation	Score	Mean	Deviation	Score
DMP1	7.496E-02	3.125E-03	0	7.484E-02	2.783E-03	0	7.571E-02	3.535E-03	0	7.505E-02	3.061E-03	0
DMP2	3.436E-03	3.500E-03	0	2.882E-03	2.477E-04	0	2.959E-03	2.749E-04	0	2.914E-03	2.301E-04	0
DMP3	2.827E-03	2.601E-04	0	2.913E-03	4.009E-04	0	2.827E-03	3.096E-04	0	2.806E-03	2.916E-04	0
DMP4	5.083E-02	1.432E-01	0	1.451E-01	2.663E-01	0	5.098E-02	1.788E-01	0	6.251E-02	1.579E-01	0
DMP5	2.179E-02	4.408E-02	0	3.103E-02	5.852E-02	0	2.773E-02	4.932E-02	0	2.179E-02	4.387E-02	0
DMP6	1.365E-02	7.114E-02	0	1.322E-02	7.121E-02	0	2.698E-03	3.340E-03	0	1.329E-02	7.092E-02	0

TABLE III
RESULTS OF EMMA2

IGDX	EMMA2-N1			EMMA2-N2			EMMA2-N3			EMMA2-N4		
	Mean	Deviation	Score	Mean	Deviation	Score	Mean	Deviation	Score	Mean	Deviation	Score
DMP1	5.471E-02	1.511E-03	0	5.475E-02	1.515E-03	0	5.461E-02	1.415E-03	0	5.401E-02	1.395E-03	1
DMP2	1.997E-03	8.976E-05	0	1.970E-03	8.781E-05	0	1.893E-03	5.193E-05	2	1.847E-03	5.311E-05	3
DMP3	1.969E-03	1.215E-04	0	1.912E-03	6.991E-05	1	1.855E-03	4.893E-05	2	1.857E-03	5.944E-05	2
DMP4	2.513E-01	2.380E-01	0	2.748E-01	2.592E-01	0	2.750E-01	2.804E-01	0	2.551E-02	1.494E-01	3
DMP5	8.416E-03	2.683E-02	0	8.431E-03	2.686E-02	0	1.767E-02	4.107E-02	0	7.935E-03	2.687E-02	3
DMP6	9.287E-04	7.199E-05	2	9.616E-04	6.691E-05	1	1.924E-03	1.104E-04	0	8.857E-04	5.069E-05	3



(a) EMMA2-N1 (b) EMMA2-N2 (c) EMMA2-N3 (d) EMMA2-N4

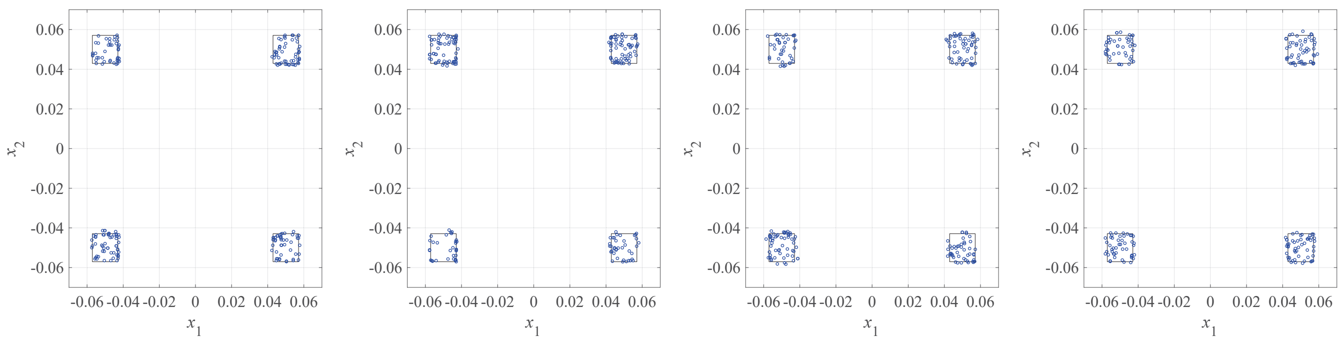
Fig. 2. The solutions obtained by EMMA2 with different normalization methods in a given single run on DMP1.

TABLE IV
THE NUMBER OF RUNS WHERE EMMA2 FIND AT LEAST ONE SOLUTION IN EVERY EQUIVALENT PARETO OPTIMAL REGION ON DMP4 AND DMP5.

Number of runs	EMMA2-N1	EMMA2-N2	EMMA2-N3	EMMA2-N4
DMP4	16	12	15	39
DMP5	38	38	34	39

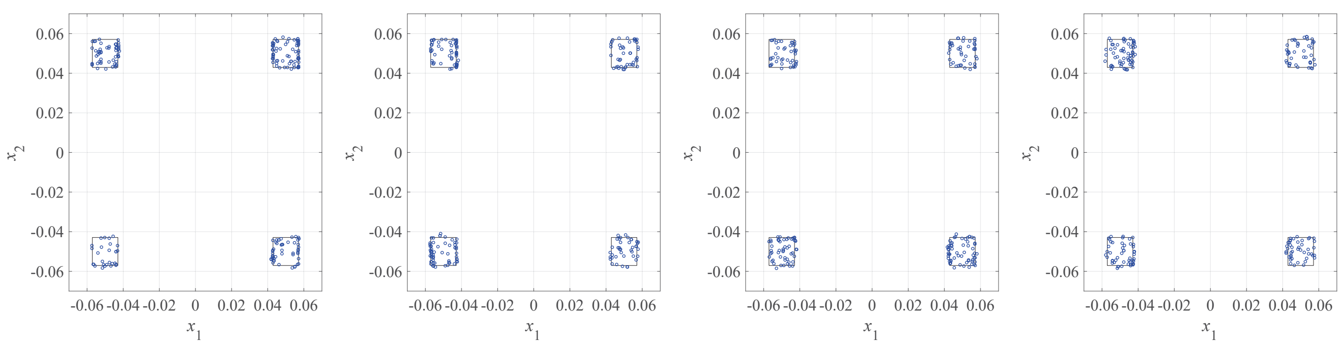
Based on the observations in Subsections IV-C1, IV-C2, and IV-C3, we can draw the following conclusions.

- Whereas EMMA2 is generally more sensitive to the choice of a normalization method, in general it performs better than EMMA1. The only difference between EMMA1 and EMMA2 is the density evaluation method in the environmental selection. That is, SCD′ is better in maintaining the diversity of solutions than SCD, and the effect of normalization is more obvious with SCD′.
- The fourth normalization method is the best in general. Different from the other three methods which only



(a) EMMA2-N1 (b) EMMA2-N2 (c) EMMA2-N3 (d) EMMA2-N4

Fig. 3. The solutions obtained by EMMA2 with different normalization methods in a given single run on DMP2.



(a) EMMA2-N1 (b) EMMA2-N2 (c) EMMA2-N3 (d) EMMA2-N4

Fig. 4. The solutions obtained by EMMA2 with different normalization methods in a given single run on DMP3.

TABLE V
COMPARISON BETWEEN EMMA1 AND EMMA2 USING PERFORMANCE SCORES.

Score	EMMA1-N1	EMMA1-N2	EMMA1-N3	EMMA1-N4	EMMA2-N1	EMMA2-N2	EMMA2-N3	EMMA2-N4
DMP1	0	0	0	0	4	4	4	5
DMP2	0	0	0	0	4	4	6	7
DMP3	0	0	0	0	4	5	6	6
DMP4	3	3	3	3	0	0	0	7
DMP5	0	0	0	0	4	4	4	7
DMP6	0	0	0	0	6	5	1	7

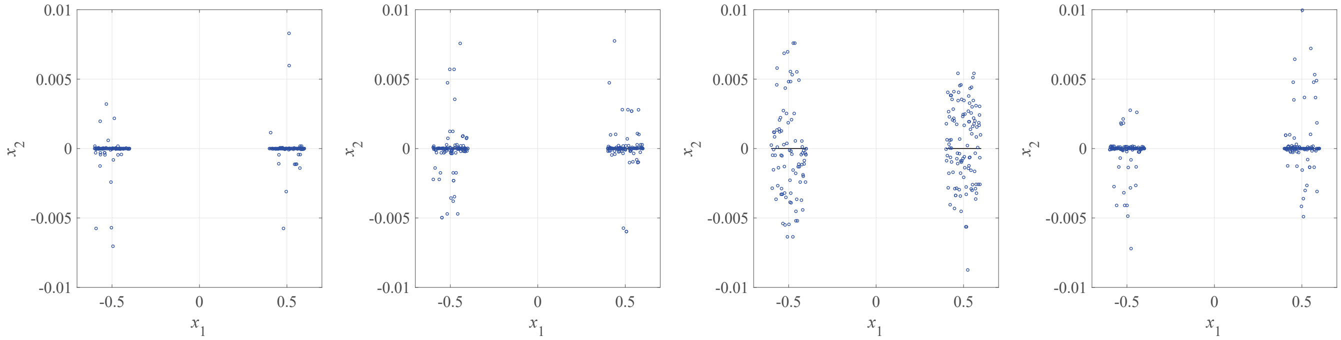
normalize the objective and decision values, the fourth method further normalizes the objective and decision space density values. The mean distance between each solution and its closet solution is able to provide proper information for normalization in most situations.

- The effect of normalization is problem-dependent. One normalization method works well on some test problems but fails on the others. For example, even though EMMA2-N4 is generally the best, it is slightly worse than EMMA2-N3 on DMP3. EMMA2-N3 also performs well on DMP2. However, it is the worst on DMP6. Thus, none of the four normalization methods can significantly outperform the others on all test problems.

V. CONCLUSION

In this study, we investigated the effect of four normalization methods on two EMMAs, i.e., EMMA1 and EMMA2. The first three methods normalize the objective and decision values in different ways. The fourth method normalizes the objective and decision space density values. EMMA1 and EMMA2 use the special crowding distance and its variant in the environmental selection, respectively.

Our experimental results on six distance minimization problems show that the effect of normalization is algorithm-dependent. EMMA1 is insensitive to normalization, whereas



(a) EMMA2-N1 (b) EMMA2-N2 (c) EMMA2-N3 (d) EMMA2-N4

Fig. 5. The solutions obtained by EMMA2 with different normalization methods in a given single run on DMP6.

the performance of EMMA2 is affected by normalization a lot. Our experimental results also show that the effect of normalization is problem-dependent. If the PS range of one decision variable is very different from that of another, the first two normalization methods may be good choices. If the PS range is very smaller than the range of feasible region, the third normalization method may be adequate. However, those normalization methods may a negative effect on the other test problems. Generally, the fourth method performs the best on most test problems.

One future work is to investigate the effect of normalization on more EMMAs on more MMOPs (such as those in [11], [13], [22]). Another is to design a strategy based on the metafeatures of MMOPs to select a good combination of EMMAs and normalization methods.

References

[1] Y. Han, D. Gong, Y. Jin, and Q. Pan, "Evolutionary multiobjective blocking lot-streaming flow shop scheduling with machine breakdowns," *IEEE Transactions on Cybernetics*, no. 99, pp. 1–14, 2017.

[2] R. Qi and G. G. Yen, "Hybrid bi-objective portfolio optimization with pre-selection strategy," *Information Sciences*, vol. 417, pp. 401–419, 2017.

[3] K. Deb and S. Tiwari, "Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1062–1087, 2008.

[4] O. M. Shir, M. Preuss, B. Naujoks, and M. T. Emmerich, "Enhancing decision space diversity in evolutionary multiobjective algorithms." in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2009, pp. 95–109.

[5] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and K. Shang, "A double-niched evolutionary algorithm and its behavior on polygon-based problems," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2018, pp. 262–273.

[6] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and Y. Han, "Searching for local pareto optimal solutions: A case study on polygon-based problems," in *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2019, pp. 896–903.

[7] C. Yue, B. Qu, and J. Liang, "A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 5, pp. 805–817, 2018.

[8] J. Liang, W. Xu, C. Yue, K. Yu, H. Song, O. D. Crisalle, and B. Qu, "Multimodal multiobjective optimization with differential evolution," *Swarm and Evolutionary Computation*, vol. 44, pp. 1028–1059, 2019.

[9] 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.

[10] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *Proceedings of IEEE World Congress on Computational Intelligence*. IEEE, 1994, pp. 82–87.

[11] Y. Liu, G. G. Yen, and D. Gong, "A multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 660–674, 2019.

[12] R. Tanabe and H. Ishibuchi, "A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2018, pp. 249–261.

[13] Y. Liu, H. Ishibuchi, N. Masuyama, and Y. Nojima, "Handling imbalance between convergence and diversity in the decision space in evolutionary multi-modal multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2019. [Online]. Available: http://dx.doi.org/10.1109/TEVC.2019.2938557

[14] H. Ishibuchi, K. Doi, and Y. Nojima, "On the effect of normalization in moea/d for multi-objective and many-objective optimization," *Complex & Intelligent Systems*, vol. 3, no. 4, pp. 279–294, 2017.

[15] L. He, Y. Nan, S. Ke, and H. Ishibuchi, "A study of the naive objective space normalization method in MOEA/D," in *Proceedings of 2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2019, pp. 1834–1840.

[16] H. Fukumoto and A. Oyama, "Impact of estimation method of ideal/nadir points on practically-constrained multi-objective optimization problems for decomposition-based multi-objective evolutionary algorithm," in *Proceedings of 2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2019, pp. 2138–2145.

[17] B. Soylu and M. Koksalan, "A favorable weight-based evolutionary algorithm for multiple criteria problems," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 2, pp. 191–205, 2009.

[18] M. Köppen and K. Yoshida, "Substitute distance assignments in NSGA-II for handling many-objective optimization problems," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 727–741.

[19] Y. Nojima, T. Fukase, Y. Liu, N. Masuyama, and H. Ishibuchi, "Constrained multiobjective distance minimization problems," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2019, pp. 586–594.

[20] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1167–1189, 2009.

[21] 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.

[22] C. Yue, B. Qu, K. Yu, J. Liang, and X. Li, "A novel scalable test problem suite for multimodal multiobjective optimization," *Swarm and Evolutionary Computation*, vol. 48, pp. 62–71, 2019.