This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

A Set-based Genetic Algorithm for Interval Many-objective Optimization Problems

Dunwei Gong, *Memeber*, *IEEE*, Jing Sun, and Zhuang Miao

***Abstract*—Interval many-objective optimization problems, involving more than three objectives and at least one subjected to interval uncertainty, are ubiquitous in real-world applications. However, there have been very few effective methods for solving these problems. In this paper, we proposed a set-based genetic algorithm to effectively solve them. The original optimization problem was first transformed into a deterministic bi-objective problem, where new objectives are hyper-volume and imprecision. A set-based Pareto dominance relation was then defined to modify the fast non-dominated sorting approach in NSGA-II. Additionally, set-based evolutionary schemes were suggested. Finally, our method was empirically evaluated on 39 benchmark interval many-objective optimization problems as well as a car cab design problem, and compared with two typical methods. The numerical results demonstrated the superiority of our method and indicated that a trade-off approximate front between convergence and uncertainty can be produced.**

***Index Terms*—genetic algorithm, many-objective optimization, uncertainty, interval, set-based evolution.**

I. Introduction

Various real-world applications can be formulated as optimization problems. They are often characterized by multiple conflicting objectives and a wide range of uncertainties that have to be taken into account (see [1]-[9]). The problems containing uncertainties and a large number of objectives, typically more than three, are termed as uncertain many-objective optimization problems. In evolutionary optimization community, uncertainty in the objective functions is generally known as noise, which is often assumed to be stochastic, and the corresponding optimization problems are termed as noisy optimization problems (see [1]-[5]). Besides random variables, uncertainty can also be viewed as intervals. Compared with the probability distribution of a random variable, the upper and the lower limits or the midpoint and the radius of an interval are easily acquired. So interval uncertainty has been widely applied in profit maximization [7], aircraft wings [8] and automobile design [9]. The optimization problems subjected to interval uncertainty are called as interval optimization problems. In this paper, we focused on interval many-objective optimization problems (IMaOPs).

Evolutionary algorithms (EAs), such as SPEA2 [10], NSGA-II [11], IBEA [12], and MOEA/D [13], are well-suited for multi-objective optimization problems (MOPs) due to their population-based nature. Particularly, a variety of methods, for example NSGA-III [14], are developed for deterministic MaOPs. These methods, however, are unsuitable for solving IMaOPs, as the objective values are intervals.

There are two approaches to solve interval MOPs by EAs: one is transforming the interval problems into deterministic MOPs (see [6], [15], [16]), and then using EAs to solve the transformed problems; the other is replacing the traditional Pareto dominance relation with interval-based ones to modify non-dominated sorting in NSGA-II (see [17]-[19]). Both methods have shortcomings. For the first one, different transformations may lead to different deterministic problems, which may generate different Pareto fronts; undoubtedly, this kind of methods may lose uncertain essence in the solving process, and thus the problem solving is imperfect. For the second one, interval-based dominance relations vary from interval order relation to interval order relation. To be more specific, a solution dominates another under an interval order relation; however, they may be non-dominated under another interval order relation. This is mainly due to the uncertainty of comparison of interval values. So it is difficult to choose a suitable dominance relation to solve interval MOPs for a user. Furthermore, for the IMaOPs, the number of non-dominated solutions will exponentially increase as the number of objectives increases, which decreases the selection pressure of Pareto-optimal solutions. Feasible approaches for solving a MaOP are utilizing new dominance relations (see [20]-[24]) to increase the selection pressure, transforming the MaOPs to MOPs by reducing the number of objectives of the optimization problem (see [25]-[29]), and so on. Obviously, these methods have difficulty in tackling IMaOPs whose objective values are intervals.

Recently, Bader *et al.* [30] and Zitzler *et al.* [31] proposed set-based MOEAs. In these methods, the individual of an evolutionary population is a solution set, the evaluation of the solution set is treated as its fitness, and the population evolves by set-based evolutionary operators. The performance indicator of a solution set is a function associated with the objective values of all solutions in the solution set, and set-based MOEAs do not involve any dominance relationship between solutions; therefore, they need not concern about the loss of selection pressure in the context of many-objective opti-

Manuscript received March 14, 2016; revised May 26, 2016, and September 5, 2016; accepted November 26, 2016. Date of publication XX XX, 2016; date of current version November 30, 2016. This work was jointly supported by National Natural Science Foundation of China (Grant Nos. 61375067, 61403155 and 61673404), National Basic Research Program of China (973 Program) (Grant No. 2014CB0463062), and Jiangsu Overseas Research & Training Program for University Prominent Young & Middle-aged Teachers and Presidents.

Dunwei Gong and Zhuang Miao are with the School of Information and Electrical Engineering, China University of Mining and Technology, Xuzhou 221008, P.R.China (e-mail: dwgong@vip.163.com; 1037407168@qq.com).

Jing Sun (Corresponding Author) is with the School of Sciences, Huaihai Institute of Technology, Lianyungang 222005, P.R.China (e-mail: sunj@hhit.edu.cn).

Digital Object Identifier XXXXXXXXX

mization. Additionally, for interval optimization problems, the performance indicator of a solution set may be deterministic, it can thus avert to compare two interval values. Above all, set-based MOEAs are well-suited for IMaOPs.

In the existing set-based MOEAs, Bader *et al.* [30] adopted hyper-volume to evaluate a solution set, and Zitzler *et al.* [31] used performance indicators with preferences to compare the two solution sets. It is well known that for interval optimization problems, at least one objective value is an interval, whose width reflects its uncertainty. The bigger the width of an interval, the more imprecise the interval, the more unstable the system to be optimized, and the more sensitive the system to uncertainty. Therefore, imprecision is an important performance indicator, suggesting that imprecision and other indicators, such as hyper-volume, should be equally treated. Under this circumstance, it is crucial to effectively distinguish two solution sets with several equal indicators. Nevertheless, a preference relation among sets proposed by Bader and Zitzler *et al.* [32] is invalid since it is based on the traditional Pareto dominance relation, and cannot break away from the curse of dimensionality.

In this paper, the IMaOP was first transformed into a deterministic bi-objective problem, where the decision variable is a solution set and two new objectives are hyper-volume and imprecision; then, through defining a set-based Pareto dominance relation and designing set-based evolutionary schemes, a set-based genetic algorithm (SetGA) was developed to solve the converted problem in the framework of NSGA-II [11].

The contributions of this paper can be summarized as follows.

(1) The IMaOP is transformed into a deterministic bi-objective problem, where objectives are hyper-volume and imprecision. The obvious character of the deterministic problem is its objectives are indicators measuring a solution set. On one hand, if using the previous method to transform the IMaOP, the objectives of the converted deterministic problem are such information as the width and the midpoint of an interval, which will multiply the number of objectives and admittedly increase the solving difficulty of the problem. As can be seen the number of objectives of the proposed method is far less than that of the previous methods. On the other hand, although some previous methods may take uncertainty into account, the information related to intervals, such as the width and the midpoint, are usually combined into one by weights (see [6], [15], [16]), which is unfavorable to understand the uncertain problem. The other advantage of the proposed method is that it can avoid comparing two intervals and the user is never confused by the issue which interval order relation should be chosen.

(2) The traditional Pareto dominance on solutions is extended on sets, and the extended dominance relation is used to compare two solution sets rather than two solutions. Furthermore, it can evidently increase the selection pressure due to the far less number of objectives of the converted problem.

(3) Uncertainty, which is extremely important in the context of interval optimization, is explicitly taken into account, and has the equal importance with convergence in interval evolutionary optimization community. As a result, the proposed method can significantly reduce uncertainty while guaranteeing convergence and diversity, which can be ascribed to hyper-volume, and thus balancing optimality and stability of a system.

(4) Interval test problems are modeled to investigate the performance of interval optimization methods. The characteristics of these interval test problems are that they are derived from well-known deterministic benchmark test problems and the Pareto-optimal front of each deterministic problem may serve as the real one of each uncertain problem; consequently, the performance indicators in the deterministic scenarios can be extended to the uncertain scenarios.

The remainder of this paper is organized as follows. Section II reviews related work. Section III proposes a method for transforming objectives of the original problem. In section IV, a SetGA suitable for the converted problem is presented. The applications of the proposed method in several benchmark IMaOPs and a car cab design problem are demonstrated in section V. Section VI concludes this paper and provides some suggestions for future research.

II. Related Work

A. EAs for interval optimization problems

Most of optimization problems suffer from uncertainty. Noise is one kind of uncertainties. The objectives of noisy optimization problems are subject to noise, which is generally viewed as a random variable; therefore, strictly speaking, noisy optimization problems in the literature are optimization problems in a stochastic environment. Under this situation, the accurate probability distribution of noise should be known *a priori*. It is, however, difficult to specify it in real-world applications. Interval optimization problems suffer from interval uncertainties and essentially are noisy ones; nevertheless, the noise in the interval optimization problems is expressed with an interval. In the application of such optimization problems, the upper and the lower limits or the midpoints and the radius of these intervals should be given in advance. It is usually easy to acquire the values of these parameters. Interval analysis [33] is a main tool for handling interval optimization problems.

We ever studied MOPs with interval uncertainties both in quantitative and qualitative objectives [6]. The problems are transformed into deterministic MOPs by using interval analysis, and then NSGA-II is adopted to solve the converted MOPs. Sahoo *et al.* [15] built a multi-objective optimization model with interval parameters where objectives are system reliability and cost. The built model was transformed into a single-objective optimization problem by a scalarizing function, and the transformed problem was solved by an improved genetic algorithm. Bhunia and Samanta [16] proposed some new definitions of interval order relation, and transformed the interval MOPs into single-objective optimization problems by different techniques. Then, the transformed problems were solved by a hybrid genetic algorithm.

It is obvious that different interval order relations may produce different deterministic optimization problems for an interval optimization problem; although some of them take uncertainty into account, the uncertain information of an

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

interval, i.e., the width and the radius, is usually combined with the lower limit or the upper limit of the interval to create a new objective, which is similar to the weighted method for MOPs. There is no denying that different methods may generate different Pareto solution sets; these sets, however, cannot fully mirror uncertain characteristic of the problem. Consequently, the above methods has some defects.

Limbourg and Aponte [17] focused on an interval MOP. Based on an order relation of intervals, a dominance relation is defined and employed to sort solutions. Accordingly, an algorithm, i.e. imprecision-propagating multi-objective evolutionary algorithm (IP-MOEA), for interval MOPs is developed. We ever studied interval MOPs as well [18]. A dominance relation and a crowding measure are defined based on interval objectives, and the method for ranking solutions is proposed. Further, We also defined a lower limit of the possibility degree, and described a dominance relation based on the above lower limit [19].

Different interval-based dominance relations may affect the sorting of solutions and produce different Pareto fronts, and choosing among them is difficult for a user with little knowledge of interval analysis. Additionally, the above-mentioned methods use imprecision as an indicator to measure an approximate front rather than a criterion to select solutions. Above all, the number of objectives is either two or three. There are no effective methods for IMaOPs so far.

B. EAs for MaOPs

Most of solutions are non-dominated in the context of many-objective optimization. Under this circumstance, it is necessary to employ new dominance relations to distinguish them so as to increase the selection pressure of Pareto-optimal solutions. In the method presented by Yang *et al.* [20], the objective space is divided into several grids and individuals are compared by using their grid coordinates. Yuan *et al.* [21] proposed a θ dominance-based evolutionary algorithm. The θ dominance relation is a strict partial order on the population. The algorithm enhances the convergence of NSGA-III by developing an efficient fitness evaluation strategy in MOEA/D in the context of many-objective optimization, and meanwhile inherits the capability of NSGA-III in maintaining the diversity. Zhang *et al.* [22] suggested an approximate non-dominated sorting algorithm for MaOPs, where the dominance relationship between two solutions is determined by performing at most three objective comparisons. In the method presented by Yang *et al.* [23], a maximum-vector-angle-first scheme is proposed to derive a well distributed and widely extended solution set, and a worse-elimination scheme is used to balance convergence and diversity. Recently, Cai *et al.* [24] utilized two distinctive components, decomposition-based-sorting and angle-based-selection, to achieve the balance between convergence and diversity.

In the method proposed by Singh *et al.* [25], Pareto corner search EA is employed to search for the corners of the Pareto front instead of the whole front. The solutions obtained by using the above algorithm are then used for dimensionality reduction to identify the relevant objectives. In addition, Saxena *et al.* [26] presented a principal component analysis and maximum variance unfolding based framework for linear and nonlinear objective reduction, respectively. Bandyopadhyay and Mukherjee [27] developed an algorithm for MaOPs, which periodically orders the objectives based on their correlation and selects a subset of conflicting objectives for further processing. In the method presented by He and Yen [28], an objective space reduction scheme and a diversity improvement strategy were used to tackle two difficulties, i.e. extremely large search space and ineffectiveness of Pareto optimality, in a framework. Cheung *et al.* [29] proposed an objective extraction method for MaOPs, which uses a linear combination of the original objectives to minimize the correlation between the reduced objectives. The above methods transform MaOPs into MOPs by dimensionality reduction; however, it is worth noting that the above reduced objectives come from the original ones.

Although the above approaches are feasible to solve MaOPs, the performance of these methods can be further improved. Additionally, they cannot solve MaOPs involving uncertainties.

C. Set-based MOEAs

Bader *et al.* [30] investigated the issue whether a multi-objective optimizer can benefit from utilizing a population of solution sets instead of relying on a single population of solutions based on hyper-volume. After designing a novel recombination operator tailored to the hyper-volume indicator, a general framework for a set-based EA is presented. Zitzler *et al.* [31] summarized recent developments in evolutionary multi-objective optimization community within a unified theory of set-based multi-objective search. They formally defined preference relations on sets and developed a specific Set-based EA. This work brings together preference articulation, algorithm design, and performance assessment under one framework and opens up a new perspective on evolutionary multi-objective optimization. Nevertheless, a hill-climbing approach is used to evolve, and different sets are ranked according to the preferences to each indicator, resulting in different importance of different indicators. Berghammer *et al.* [34] took an order-theoretic view on the convergence of set-based multi-objective optimization, and examined that the use of set indicators can help to direct the search towards Pareto-optimal sets. They pointed out that the above dominance relation defined on sets may lead to a cyclic behavior, i.e., the algorithm may return to a set dominated by others which are obtained during the previous evolution. Rudolph [35] proved the existence of set-based EAs, whose population converges to an approximate front with the maximal hyper-volume for a given reference point, by providing a concrete example that converges geometrically fast. We ever suggested a set evolution guided PSO for MaOPs [36]. The MaOP is first converted into a bi-objective optimization problem, where objectives are distribution and hypervolume; then, the solution sets are regarded as particles; finally, a particle updating strategy for sets is given and a method of selecting the best set particle for updating sets is developed.

The above approaches either use a single indicator [30] or several indicators with preferences [31], or a weighted

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

indicator [36] to evaluate a solution set, which implies that a MOP is degraded to a single-objective optimization problem, or various indicators are addressed by a hierarchic process. The importance of different indicators is unequal.

In order to highlight the importance of uncertainty, the indicators of imprecision and hyper-volume were adopted to transform an IMaOP into a deterministic bi-objective problem, and a SetGA was developed to simultaneously optimize the two objectives in this paper. It is worth mentioning that the proposed transformation method is totally different from any of the previous methods. Firstly, it is completely different from the methods which transform uncertain optimization problems into deterministic ones by midpoint, width, and so on, since it obviously reduces the number of objectives and explicitly dealing with uncertainty. Secondly, it is greatly different from the previous dimensionality reduction methods, because the converted objectives are new rather than a subset of the original objectives and the decision variable of the converted problem is a solution set of the original problem rather than a solution. Finally, it is obviously different from the previous set-based methods, as the importance of imprecision is emphasized, which means that hyper-volume and imprecision are simultaneously optimized, and they have the same position. To sum up, the proposed transformation method has the following superiorities: (1) obviously reducing the number of objectives, (2) helping the user out of trouble for choosing an appropriate interval-based dominance relation, (3) and greatly reducing uncertainty while ensuring convergence and diversity.

In the proposed method, transforming the original objectives based on indicators, defining a set-based Pareto dominance relation and designing set-based evolutionary schemes are pivotal techniques to be solved. In the next section, the approach of transforming the original objectives will be given.

III. Transforming Objectives

Consider the following optimization problem:

$$\begin{aligned}&\min \mathbf{f} = (f_1(\mathbf{x}, \mathbf{c}_1), f_2(\mathbf{x}, \mathbf{c}_2), \cdots, f_m(\mathbf{x}, \mathbf{c}_m))^T,\\ &\text{s.t.} \quad \mathbf{x} \in \mathbf{D} \subseteq \mathbf{R}^n,\\ &\mathbf{c}_i = (c_{i1}, c_{i2}, \cdots, c_{il})^T, c_{ik} = [\underline{c}_{ik}, \overline{c}_{ik}], k = 1, 2, \cdots, l,\end{aligned} \tag{1}$$

where $\mathbf{x}$ is an n-dimensional decision variable; $\mathbf{D}$ is the decision space of $\mathbf{x}$; $f_i(\mathbf{x}, \mathbf{c}_i) \triangleq [\underline{f}_i(\mathbf{x}, \mathbf{c}_i), \overline{f}_i(\mathbf{x}, \mathbf{c}_i)]$ is the ith objective function with interval parameters, $i = 1, 2, \cdots, m$; m is the number of objectives, and $m > 3$; $\mathbf{c}_i$ is a fixed interval vector parameter, and independent to the variable, $\mathbf{x}$, suggesting that it remains unchanged along with $\mathbf{x}$; $\mathbf{c}_{ik} = [\underline{\mathbf{c}}_{ik}, \overline{\mathbf{c}}_{ik}]$ is the kth component of $\mathbf{c}_i$.

It is worth noting that an interval is also a set, consisting of the components larger than its lower limit and smaller than its upper limit, and differs from the afore-mentioned solution set. The set in this paper specially refers to the solution set.

For any two solutions, $\mathbf{x_1}, \mathbf{x_2} \in \mathbf{D}$, of problem (1), the corresponding ith objectives are $f_i(\mathbf{x}_1, \mathbf{c}_i)$ and $f_i(\mathbf{x}_2, \mathbf{c}_i)$, $i = 1, 2, \cdots, m$, respectively. Limbourg and Aponte [17] defined the following order relation between them.

Definition 1[17] $<_{IN}$ is defined as:

$$\begin{aligned}&f_i(\mathbf{x}_1, \mathbf{c}_i) <_{IN} f_i(\mathbf{x}_2, \mathbf{c}_i) \Leftrightarrow \underline{f}_i(\mathbf{x}_1, \mathbf{c}_i) \le \underline{f}_i(\mathbf{x}_2, \mathbf{c}_i)\\ &\wedge \bar{f}_i(\mathbf{x}_1, \mathbf{c}_i) \le \bar{f}_i(\mathbf{x}_2, \mathbf{c}_i) \wedge f_i(\mathbf{x}_1, \mathbf{c}_i) \ne f_i(\mathbf{x}_2, \mathbf{c}_i).\end{aligned}$$

Otherwise, two objective intervals are incomparable, denoted as $f_i(\mathbf{x}_1, \mathbf{c}) \| f_i(\mathbf{x}_2, \mathbf{c})$.

The order relation $<_{IN}$ is antisymmetric, reflexive and transitive, and hence defines a partial order relation between intervals.

Further, the dominance relation based on interval is defined as follows.

Definition 2[17]

$$\begin{aligned}&\mathbf{x}_1 \succ_{IP} \mathbf{x}_2 \Leftrightarrow\\ &\forall i \in \{1, 2, \ldots, m\}, f_i(\mathbf{x}_1, \mathbf{c}_i) <_{IN} f_i(\mathbf{x}_2, \mathbf{c}_i)\\ &\qquad\qquad \vee f_i(\mathbf{x}_1, \mathbf{c}_i) \| f_i(\mathbf{x}_2, \mathbf{c}_i)\ ,\\ &\exists i \in \{1, 2, \ldots, m\}, f_i(\mathbf{x}_1, \mathbf{c}_i) <_{IN} f_i(\mathbf{x}_2, \mathbf{c}_i).\end{aligned}$$

This section focuses on the transformation of the objectives of problem (1). State-of-the-art set indicators [37] fall into three categories according to convergence, diversity, and spread of a Pareto front. There is another important category, uncertainty, for interval MOPs. If choosing an indicator as a novel objective from each category, the number of objectives of the converted problem will be obviously less than that of problem (1). Under this situation, the variable of the converted problem is a solution set of problem (1). Furthermore, although the objectives of problem (1) are uncertain, the value of an indicator may be exact. As a result, the objective values of the converted problem may be exact as well. Another benefit of the transformation is that it does not need to use any interval-based dominance relation.

In the light of the above discussion, appropriate indicators were employed to transform problem (1) into a deterministic MOP in this paper. Compared with existing approaches of the dimensionality reduction, the advantages of the proposed approach are as follows: (1) the objectives of the converted problem are closely related to the quality of an approximate front of the original problem, neither a part of objectives of the original problem nor their aggregation; (2) the variable of the converted problem is a solution set of the original problem rather than its solution. As a result, the proposed approach is distinctly different from the existing approaches.

There exist two indicators, i.e., hyper-volume and imprecision proposed by Limbourg and Aponte [17], in the context of interval multi-objective optimization.

Definition 3[17] For an approximate Pareto-optimal solution set, X, of problem (1), the hyper-volume is defined as

$$\begin{aligned}&H(X) = \left[\underline{H(X)}, \overline{H(X)}\right] =\\ &\wedge\left(\bigcup_{\mathbf{x}\in X}\{\mathbf{y} \in R^n \,|\mathbf{x}_{ref} \prec_{IP} \mathbf{y}_{IP} \prec \mathbf{x}\}\right),\end{aligned}$$

where $\mathbf{x}_{ref}$ is a reference point; $\wedge$ is Lebesgue measure; $\underline{H(X)}$ and $\overline{H(X)}$ are the worst-case and the best-case hyper-volume, respectively. The hyper-volume rewards convergence towards the true front as well as representative distribution of points along the front [38], and is the unique indicator known to be strictly monotonic w.r.t. traditional Pareto dominance relation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

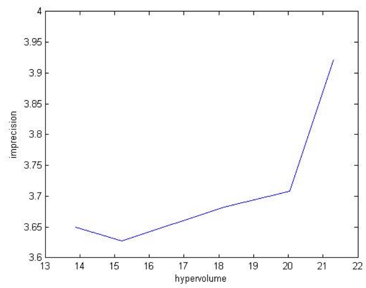

Fig. 1. The curve of imprecision w.r.t. hypervolume

[32]. Consequently, the bigger the value of hyper-volume of an approximate front, the better its performance.

Imprecision proposed by Limbourg and Aponte [17] measures uncertainty of an approximate front and is represented by the added volumes of all hyper-cubes in a front. The magnitude of the volume of hyper-cube will reduce sharply with the increase of objectives; therefore, this representation is unsuitable for IMaOPs. The following imprecision is proposed in this paper.

Definition 4 For an approximate Pareto-optimal solution set, X, of problem (1), the imprecision of the front corresponding to X is defined as follows:

$$I(X) = \sum_{\mathbf{x} \in X} \sum_{i=1}^{m} \left(\overline{f}_i(\mathbf{x}, \mathbf{c}_i) - \underline{f}_i(\mathbf{x}, \mathbf{c}_i) \right). \tag{2}$$

The above imprecision is represented by the added half of perimeters of all hyper-cubes in the front. The smaller the value of the imprecision, the smaller the uncertainty of the front.

As stated above, hyper-volume can show the convergence and the diversity of an approximate front, and imprecision can reflect its uncertainty. Hyper-volume and imprecision do not conflict with each other sometimes. However, when the population approaches the front, a big value of hyper-volume usually leads to a big value of imprecision, just as shown in Fig. 1. Fig. 1 plots the curve of imprecision w.r.t. hypervolume when using IPMOEA to tackle bi-objective ZDT1, and visually reveals that with the increase of the hypervolume, the corresponding imprecision zigzag increases, suggesting that hyper-volume and imprecision partly conflict with each other. This characteristic makes the proposed set-based method significant reduce uncertainty while ensuring convergence and achieve a competitive performance, which has been demonstrated in our experiments. In this paper, problem (1) was transformed into the following deterministic bi-objective optimization problem by using the worst-case hyper-volume and imprecision:

$$\begin{aligned} &\max \underline{H(X)}, \\ &\min \overline{I(X)}, \\ &\text{s.t.} \quad X \in 2^{\mathbf{D}}, \end{aligned} \tag{3}$$

where $2^{\mathbf{D}}$ is the power set of $\mathbf{D}$ which includes all subsets of $\mathbf{D}$, i.e., $2^{\mathbf{D}} = \{X \,|X \subseteq \mathbf{D}\}$; X is a solution set of problem (1).

Problem (3) is a bi-objective optimization problem, where the first objective is to be maximized, while the second is to be minimized. The above problem can be further transformed into the following bi-objective maximization problem by multiplying the second objective by -1:

$$\begin{aligned} &\max F(X) \stackrel{\Delta}{=} (F_1(X), F_2(X)) = (\underline{H(X)}, -I(X)), \\ &\text{s.t.} \quad X \in 2^{\mathbf{D}}. \end{aligned} \tag{4}$$

The optimal solutions of problem (4) are just trade-off approximate Pareto-optimal solution sets between convergence and uncertainty of problem (1), and the optimality and stability of a system are consequently balanced.

If using an evolutionary method to solve problem (4), each individual of the evolutionary population will be a set rather than an ordinary solution of problem (1), suggesting that traditional MOEAs are unsuitable for solving problem (4). So a SetGA tackling the above problem was developed in this paper, which will be described in the next section. The difference from traditional individuals is that individuals in SetGAs are called set-based individuals.

IV. SetGA

This section describes a SetGA for solving problem (4). The main idea of the method is as follows: in the framework of NSGA-II [11] procedure, the non-dominated sorting approach is modified by a set-based dominance relation, and set-based evolutionary schemes are developed. The set-based dominance relation and evolutionary schemes, which will be elaborated in subsection IV.A and IV.B, respectively, are two key techniques of the methodology. Its framework, complexity, and comparison with IP-MOEA [17] will be stated in subsection IV.C.

A. Set-based Dominance

This subsection addresses the issue of distinguishing different sets, i.e., solutions of the converted problem. Bader and Zitzler *et al.* [32] proposed a preference relation among sets on the basis of the traditional Pareto dominance relation. The following definition is derived by applying the above preference relation to problem (1).

Definition 5 [32] For any two solutions, $X_1, X_2 \in 2^{\mathbf{D}}$, of problem (4), the set preference relation on $2^{\mathbf{D}}$ is defined as follows:

$$X_1 \succ X_2 \Leftrightarrow \forall b \in X_2, \exists a \in X_1, a \succ_{IP} b. \tag{5}$$

The essence of the above set preference relation is the comparison of solutions belonging to different sets. Therefore, the number of non-dominated sets of problem (4) will increase along with that of the objectives of problem (1), resulting in loss of the selection pressure of problem (4).

The number of the objectives of problem (4) is far less than that of problem (1). If employing the objectives of problem (4) to distinguish different sets, non-dominated sets will be obviously decreased; accordingly, the selection pressure of problem (4) will be increased. The following set-based dominance relation was thus presented.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

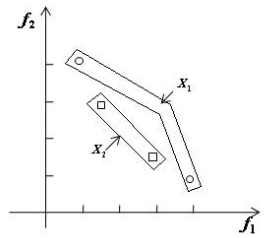

Fig. 2. Illustration of definition 6

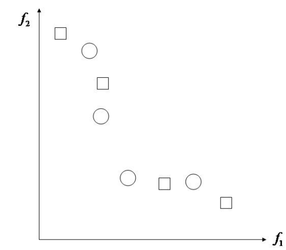

Fig. 3. Illustration of set-based individuals

Definition 6 For any two solutions, $X_1, X_2 \in 2^{\mathbf{D}}$, of problem (4), if $F_1(X_1) \geq F_1(X_2)$, $F_2(X_1) \geq F_2(X_2)$, and at least one greater-than relation is held, X_1 is said to dominate X_2 based on set, denoted as $X_1 \succ_{SetP} X_2$.

The above definition is an extension of the traditional Pareto dominance relation, and can also be generalized to an arbitrary MOP whose variable is a set.

The following example illustrates the difference between definitions 5 and 6. Assume that the original problem is a deterministic bi-objective maximization problem, and the sets of objective values corresponding to two solution sets, X_1 and X_2, are $\{(1,4),(4,1)\}$ and $\{(3,1.5),(1.5,3)\}$, respectively, as shown in Fig. 2, where circles and boxes represent X_1 and X_2, respectively. X_1 and X_2 are non-dominated sets by definition 5. However, when converting the original problem into another bi-objective problem by using hyper-volume and spread indicators proposed by Zitzler [39], where the reference point is $(0,0)$, the objective values corresponding to X_1 and X_2 are $(7, 3\sqrt{2})$ and $(6.75, 1.5\sqrt{2})$, respectively. X_1 dominates X_2 based on set by definition 6. So the proposed set-based dominance relation can effectively compare the qualities of sets, and thus increasing the selection pressure of problem (4).

B. Set-based Evolutionary Schemes

This subsection clarifies set-based genetic operators, including mating selection, crossover, mutation, and environmental selection, where mating and environmental selections are based on a sequence of evolutionary individuals, so the method for sorting individuals is first introduced.

1) Sorting: The objects of set-based sorting are sets rather than solutions, so there are two difficulties: one is non-dominated sorting tailored to set-based individuals, and the other is to further distinguish the set-based individuals with the same rank. To this end, the set-based dominance relation, proposed in subsection IV.A, was adopted to sort set-based individuals of an evolutionary population instead of the traditional Pareto dominance relation in NSGA-II [11]. Generally, there are several individuals with the same rank. In order to distinguish them, Deb *et al.* [11] employed the crowding distance to reflect the diversity of individuals; in addition, Bader and Zitzler [32] utilized the contribution of an individual to hyper-volume to maximize hyper-volume of an approximate front. Nevertheless, it is difficult to measure the distance between set-based individuals which may blend, shown as X_1 and X_2 in Fig. 3. Additionally, there are four categories of indicators for interval MOPs, and the objectives of problem (4) show the convergence, diversity and uncertainty of an approximate front. Therefore, a spread metric was defined to measure extension of an approximate front. It is worth mentioning that the objective values of an individual are real for the spread metric proposed by Zitzler [39], so the above spread is unsuitable since the objective values of problem (1) are intervals, suggesting that a novel spread metric should be defined in the context of interval uncertainty.

Definition 7 For any set-based individual in an evolutionary population, its spread metric, denoted as $Spread(X)$, is defined as follows:

$$Spread(X) = \sqrt{\sum_{i=1}^{m} \left(\max_{\mathbf{x} \in X} \overline{f}_i(\mathbf{x}, \mathbf{c}_i) - \min_{\mathbf{x} \in X} \underline{f}_i(\mathbf{x}, \mathbf{c}_i) \right)^2}, \tag{6}$$

where $\mathbf{x}$ is a solution in X. The larger the value of $Spread(X)$, the better the spread of X, and the broader the distribution of its corresponding front.

It can be noted that in definition 7, the maximum and the minimum of each objective in the spread metric proposed by Zitzler [39] are replaced with the maximum value of the upper limits and the minimum one of the lower limits for each objective interval, respectively.

It is worth noting that formula (6) is different from formula (2). $\overline{f}_i(\mathbf{x}, \mathbf{c}_i) - \underline{f}_i(\mathbf{x}, \mathbf{c}_i)$ in formula (2) is the width of the ith objective interval, which represents uncertainty of an individual on the ith objective. Whereas, $\max_{\mathbf{x} \in X} \overline{f}_i(\mathbf{x}, \mathbf{c}_i) - \min_{\mathbf{x} \in X} \underline{f}_i(\mathbf{x}, \mathbf{c}_i)$ in formula (6) is the difference between the smallest lower limit and the biggest upper limit of all the ith objective intervals, which represents the extension of all individuals on the ith objective. As a consequence, they do not conflict with each other.

The following scheme was adopted to sort set-based individuals. The set-based individuals in the evolutionary population are first sorted by the set-based dominance relation; then those with the same rank are sorted in a descending order according to their spread metrics. The smaller the rank of a set-based individual and the larger the value of its spread, the better its performance. The detail of the procedure is given by Algorithm 1.

2) Mating Selection: Tournament selection of size two was adopted. The detailed scheme is described as follows. For any two set-based individuals, X_1 and X_2, if X_1 dominates X_2 based on set, i.e., $X_1 \succ_{SetP} X_2$, select X_1; if X_2 dominates X_1 based on set, i.e., $X_2 \succ_{SetP} X_1$, select X_2; otherwise,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

Algorithm 1 Sorting

Require: Population P

procedure $setSort(P)$
$[F_1, \cdots, F_l]$; % Population P is sorted by set-based pareto dominance.
for i=1 to l **do**
  for all $X \in F_i$ **do**
    Calculating $spread(X)$; % Calculating the spread of a set-based individual
  end for
  $F_i \leftarrow sort(F_i)$; % Population P is sorted according to the spread.
end for
$P \leftarrow [F_1, \cdots, F_l]$;
end procedure

select the one with a larger value of spread. The detail of the procedure is illustrated by Algorithm 2.

Algorithm 2 Mating Selection

Require: Population P

procedure $setMatingSelect(P)$
for k=1 to N **do**
  repeat
  $i \leftarrow random[1, N]$;
  $j \leftarrow random[1, N]$;
  until $(i \neq j)$
  if $P_i \prec_{SetP} P_j$ **then**
    $MP_k \leftarrow P_i$;
    elseif $P_j \prec_{SetP} P_i$ **then**
    $MP_k \leftarrow P_j$;
    else
    if $Spread(P_i) > Spread(P_j)$ **then**
      $MP_k \leftarrow P_i$;
      else
      $MP_k \leftarrow P_j$;
    end if
  end if
end for
end procedure

3) Crossover: A crossover operator within a set-based individual, called interior crossover, was considered in this paper, which is special for set-based evolution. For a set-based individual, its interior crossover is equivalent to crossover between different components within it, and is similar to the crossover operator in a traditional GA. An arbitrary crossover operator can be adopted, and simulated binary crossover was employed in our experiments. The detail of the procedure is given by Algorithm 3. It is of necessity to note that crossover between two set-based individuals is more challenging, which will be our future research topic.

4) Mutation: Mutation of one or more solutions within a set-based individual is called set-based mutation. Fig.4 illustrates a set-based mutation operator, implying that any ordinary mutation operator, e.g., single-point and polynomial mutation,

Algorithm 3 Crossover

Require: a set-based individual X

procedure $setCrossover(X)$
$Y = \emptyset$
repeat
repeat
$i \leftarrow random[1, \omega]$;
$j \leftarrow random[1, \omega]$;
until $(i \neq j)$
$Y \cup crossover\,(\mathbf{x}_i, \mathbf{x}_j)$; % $crossover()$ is a solution-based crossover.
until $(size(Y) = \omega)$
end procedure

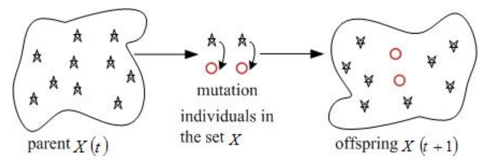

Fig. 4. Set-based mutation operator

can be employed in SetGA, and polynomial mutation was used in our experiments. The detail of the procedure is given by Algorithm 4.

Algorithm 4 Mutation

Require: a set-based individual X

procedure $setMutate(X)$
for i=1 to ω **do**
  $\mathbf{x}_i \leftarrow mutate(\mathbf{x}_i)$; % $mutate()$ is a solution-based mutation.
end for
end procedure

5) Environmental Selection: The $(\mu + \mu)$ scheme was utilized to replace the current population. To this end, the parent and its offspring populations are combined together to form one whose individuals are ranked in terms of the proposed sorting scheme, from which the first μ individuals are selected to form the next population. The details of the procedure is given by Algorithm 5.

Algorithm 5 Environmental selection

Require: Parent $P(t)$ and offspring $Q(t)$

procedure $setEnviornmentalSelect(P(t), Q(t))$
$P(t+1) \leftarrow setSort(P(t) \bigcup Q(t))[1, N]$;
end procedure

C. Steps of SetGA

In the light of the set-based evolutionary schemes in subsection IV.B, the detailed steps of SetGA are as follows.

Step 1 Initialize a population, $P(0)$, of size N whose individual contains ω solutions of problem (1); sort it by *setSort*$(P(0))$, and set $t = 0$;

Step 2 Select set-based individuals by *setMatingSelect*($P(t)$); perform *setCrossover* ($P(t)$) and *setMutate*($P(t)$) to create an offspring population, $Q(t)$, of size N;

Step 3 Use the proposed environmental selection operator, shown as Algorithm 5, to produce the next population, $P(t+1)$;

Step 4 Judge whether the termination criterion is met or not. If yes, output the optimal solutions; otherwise, set $t=t+1$, and go to Step 2.

Since the objectives of problem (4) are hyper-volume and imprecision, whose values are closely related to the number of solutions in a set-based individual, and different numbers may lead to different objective values of problem (4), which is unfavorable to fairly compare the set-based individuals, the numbers of solutions in all the set-based individuals are equal.

The complexity of an iteration of SetGA was analyzed. The basic operators and their complexities in the worst case are as follows.

(1) Fitness calculation is $O(2N\omega^m)$, given that the complexity of calculating imprecision can be omitted, and that calculating hyper-volume of each set-based individual is $O(\omega^m + m\omega log\omega)$ using the method proposed by Bader and Zitzler [32].

(2) Non-dominated sorting of the combined population $P(t)\bigcup Q(t)$ is $O(8N^2)$.

(3) Spread metric assignment is $O(4m\omega N)$, owing to searching for the maximum and the minimum from the upper and the lower limits of ω objective values, respectively, in m objectives.

(4) Spread metric sorting is $O(2Nlog(2N))$.

So the overall complexity of SetGA is $O(N\omega^m + N^2)$, which is governed by the fitness calculation.

When analyzing the complexity of IP-MOEA, the population size is set to ωN. The complexity of IP-MOEA can be inferred as $O(\omega^m N^m)$ by using the same method.

The following conclusion can be drawn from the above analysis: the complexity of SetGA is smaller than that of IP-MOEA, and CPU time of SetGA will thus be distinctly less than that of IP-MOEA along with the number of objectives. The possible reason is that although the complexities of both SetGA and IP-MOEA are governed by the hyper-volume calculation, the former is determined by the number of solutions of problem (1) within an individual, whereas the latter depends on the number of solutions within a population. The number of solutions within an individual is far smaller than that within a population.

V. Experiments

The performances of the proposed method were evaluated by optimizing 39 benchmark IMaOPs and a problem from practice, and by comparing it with a typical interval MOEA, i.e. IP-MOEA [17], and a method which utilized NSGA-III [14] to optimize the midpoint of an objective interval, called NSGA-IIIR. The implementation environment was as follows: Inter(R) Core(TM) i3-3240 CPU, 4.00GB RAM, windows 7 and Matlab R2010a. Each method was run 20 times independently, and the averages of these results were calculated.

A. Test Problems

1) Deterministic Test Problems: The DTLZ [40] and the WFG [41] test problem suites were invoked since the number of their objectives are scalable, here, ranging from 5 to 20 objectives. The number of their variables is $n=k+l$, where k and l are called as position- and distance-related parameters, respectively [41]. For the DTLZ problems, DTLZ1-DTLZ7, k is always fixed and related to the number of objectives, and $k=m-1$ in our experiments; according to the settings proposed by Deb *et al.*, $l=5$ was used in DTLZ1, and $l=10$ in DTLZ2-DTLZ7. For the WFG problems, WFG1-WFG6, k must be divisible by $m-1$, and $k=m-1$ in our experiments; l can be set to any positive integer, except for WFG2 and WFG3, for which l must be a multiple of two; $l=6$ was used for WFG2 and WFG3, and $l=5$ for others in our experiments. Additionally, a car cab design proposed by Deb and Jain [42] was considered. It is a 9-objective optimization problem with 11 decision variables involving dimensions of the car body and bounds on natural frequencies, whose objectives are related to roominess of the car, fuel economy, acceleration time, and road noise at different speeds.

2) Modeling Interval Test Problems: There are two approaches to convert the above deterministic test problems into interval ones. One is introducing an imprecision factor ϵ in the objective space, just like the following method proposed by Limbourg and Aponte [17].

$$\begin{aligned} \underline{f}_i(\mathbf{x},\epsilon_i) &= 1-\max\{f_i(\mathbf{x}), f_i(\mathbf{x})+\epsilon_i\}, \\ \overline{f}_i(\mathbf{x},\epsilon_i) &= 1-\min\{f_i(\mathbf{x}), f_i(\mathbf{x})+\epsilon_i\}, \end{aligned} \tag{7}$$

where $[\underline{f}_i(\mathbf{x},\epsilon_i), \overline{f}_i(\mathbf{x},\epsilon_i)]$ is an interval objective function of $\mathbf{x}$, $f_i(\mathbf{x})$, $i=1,2$, is the real value of objective function, and $\epsilon=(\epsilon_1,\epsilon_2)=(\sin(10\pi\sum_i x_i)/2, \sin(20\pi\sum_i x_i)/2)$. Another is introducing interval parameters, which is widely used in the context of interval optimization.

However, if we change the deterministic parameters of benchmark problems, namely ZDT, DTLZ and WFG test problem suits, into intervals, the creation of test problems will be extremely complicated, the characteristics of Pareto-optimal front, such as convex, concave, linear and disconnected geometries, may be damaged, and furthermore, it is difficult to determine the true Pareto-optimal front of each test problem. Therefore, we adopted the first method to transform DTLZ and WFG test problem suits into interval test problems in this paper. The biggest benefit is that the Pareto-optimal front of each deterministic benchmark test problem may serve as the real one of each uncertain problem; consequently, the performance indicators in the deterministic scenarios, including generation distance (GD), inverse generation distance (IGD), and so on, can be extended to the uncertain scenarios. Nevertheless, ϵ introduced by Limbourg and Aponte [17] is only for bi-objective ZDT test problem suit, and is in the range of -1 and 1 for any objective. This assumption is unreasonable in most uncertain scenarios, as higher objective values are usually expected to have more errors than lower ones, which implies that the uncertainty is always related to both the variables and the corresponding objective value.

To model the interval uncertainty more reasonably and more accurately, it is assumed that (1) the real objective values of deterministic test problems are the middles of objective intervals of corresponding interval test problems; (2) the uncertainty is related to both the variables and the real objective value. In real-world applications, the less the standard deviation σ, the more stable the uncertain data, and the higher the accuracy of the uncertain data. So the standard deviation of uncertain data is not too big. According to the experimental methods in most noisy optimization problems, the standard deviation is set as 0.2, which is a higher noise level in the context of noisy optimization; (3) it is rather reliable to use an interval to express uncertain data.

According to assumption 2), the standard deviation σ is suggested as follows:

$$\sigma_i = \lambda \left| \sin \left(10 i \pi \sum_j x_j \right) \right| f_i(\mathbf{x}), \tag{8}$$

where λ is the highest relative standard deviation, and is set as 0.1 in our experiments; i denotes the ith objective.

According to assumptions 1) and 3), provided that the objectives of deterministic optimization problem are denoted as $f_i(\mathbf{x}), i = 1, 2, \cdots, m$, interval objectives, denoted as $f_i(\mathbf{x}, \sigma_i)$, were defined as follows:

$$f_i(\mathbf{x}, \sigma_i) = \left(f_i(\mathbf{x}) - \sigma_i, f_i(\mathbf{x}) + \sigma_i \right). \tag{9}$$

Since the objective values of $f_i(\mathbf{x}, \sigma_i)$ defined by formula (9) are intervals, the objectives of the original deterministic optimization problems were transformed into interval objectives. The resulted optimization problems were denoted as DTLZ$_I$1-DTLZ$_I$7 and WFG$_I$1-WFG$_I$6.

For the car cab design problem, the second approach was employed to make the objective values intervals. The resulted optimization problems were denoted as Car$_I$.

B. Parameter Settings

For IP-MOEA, the population sizes were set to 200 for 5-objective optimization problems, and 500 for 15- and 20-objective problems; tournament selection, simulated binary crossover and polynomial mutation operators were employed. In SetGA, the population size and the number of solutions within a set-based individual were set to 4 and 50, respectively for 5-objective optimization problems and car cab design problem, and they were set to 5 and 100, respectively, for 15- and 20-objective problems; in addition, set-based evolutionary schemes proposed in subsection IV.B were adopted. For the two methods, the crossover and mutation probabilities were set to 0.9 and 0.1, respectively, the distribution indexes for crossover and mutation operators were 20, and the maximal number of generations was 200 for all optimization problems. A python experimental platform [43] was used to implement NSGA-IIIR. The total function evaluations of SetGA and IP-MOEA were 40000 for 5-objective optimization problems, and 100000 for 15- and 20-objective problems; the function evaluations for each optimization problem, which was determined by the platform, were listed in Table I, except for the car cab design problem, which was not in the platform. Monte Carlo simulation [32] was used to approximate the exact value of hyper-volume, where the number of sampling points was 10000.

TABLE I
FUNCTION EVALUATIONS IN NSGA-IIIR

	5D	15D	20D
DTLZ$_I$1	100096	100080	100064
DTLZ$_I$2	30080	30000	30104
DTLZ$_I$3	100096	100080	100064
DTLZ$_I$4	30080	30000	30104
DTLZ$_I$5	30080	30000	30104
DTLZ$_I$6	100096	100080	100064
DTLZ$_I$7	30080	30000	30104
WFG$_I$1	30080	30080	30080
WFG$_I$2	30080	30080	30080
WFG$_I$3	30080	30080	30080
WFG$_I$4	30080	30080	30080
WFG$_I$5	30080	30080	30080
WFG$_I$6	30080	30080	30080

C. Performance Indicators

The following five indicators were employed to compare the performance of different methods:

(1) Worst-case hyper-volume [17] (H metric, for short). The larger the value of H metric of the final front obtained by a method, the closer the final front to the true front, and the better the distribution of the final front obtained by the method.

(2) Imprecision (I metric, for short). The smaller the value of I metric of the final front obtained by a method, the more precise the final front obtained by the method.

(3) Spread metric (S metric, for short).

(4) IGD. The Pareto-optimal front of each deterministic benchmark test problem is used as the real one of each uncertain problem, and the reference set comes from the python experimental platform. The IGD metric is calculated as the distance between the reference set and a set constituted by the upper limits of the objective values of the final Pareto-optimal solution set.

(5) CPU time (T metric, for short). The smaller the value of T metric of a method, the higher the efficiency of the method.

It needs to point out that in our experiments, when the termination criterion of SetGA and IP-MOEA [17] is met, we seek the Pareto front of the final population, which is constituted by the solutions in all set-based individuals, and then the values of the four performance indicators of the obtained Pareto front are calculated. While NSGA-IIIIR ends, it will produce a final Pareto-optimal solution set; we calculate the values of imprecision, spread, the worst hyper-volume and IGD of the Pareto front which is constituted by objective intervals of problem (1) corresponding to the final solution set.

D. Experimental Results

The experiments were divided into the following two groups: the first was used to validate the superiority of the proposed set-based Pareto dominance relation through the proportion of non-dominated solutions in a population, and the second compared the performances of SetGA with IP-MOEA and NSGA-IIIR.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

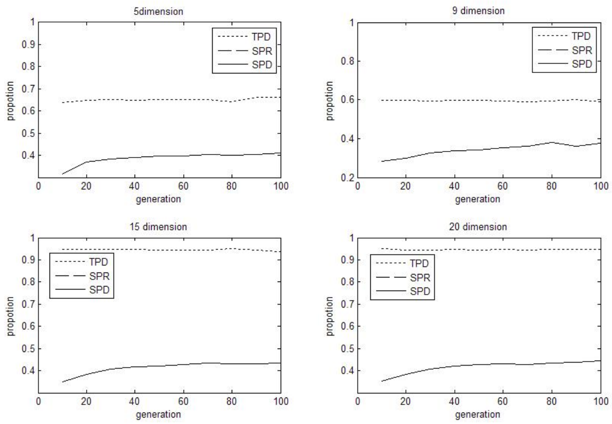

Fig. 5. Average proportions of non-dominated solutions w.r.t. number of generations for different dimensional optimization problems

1) Performances of different dominance relations: This group of experiments investigated the performances of the set-based Pareto dominance relation proposed by definition 6 in this paper, called SPD, and two previous dominance relations. One is an interval-based Pareto dominance relation proposed by Limbourg and Aponte [17], defined as definition 2 and called TPD, and the other is the set preference relation presented by Bader and Zitzler [32], defined as definition 5 and called SPR.

The proportions of non-dominated individuals in the evolutionary population during the evolution of SetGA and IP-MOEA were taken into account to compare the performances of SPD and TPD. To investigate the performances of SPD and SPR, two SetGAs were utilized to obtain the proportions of non-dominated solutions during the evolution. Except that they adopted different dominance relations, the same genetic operators were employed.

Fig. 4 depicts the curves of the average proportions of non-dominated solutions during the evolution when using different dominance relations to solve different dimensional optimization problems. Since the proportions of non-dominated solutions corresponding to SPR are 1 for all optimization problems, the dashed curves and the top borders of the figures are overlapped.

It is clear from Fig. 4 that:

(1) For the same generations, the proportion of non-dominated solutions of SetGA is distinctly smaller than the others, indicating that SPD can obviously increase the selection pressure of problem (4), as expected in subsection IV.A.

(2) Along with the evolution of a population, individuals corresponding to SPR are non-dominated for each optimization problem, suggesting that SPR is unfavorable to distinguish the individuals; those corresponding to TPD are nearly non-dominated for 15- and 20-objective optimization problems, implying that in the context of many-objective optimization, TPD cannot distinguish the individuals well; nevertheless, the proportion of non-dominated solutions corresponding to SPD is smaller than 0.5 for each optimization problem, indicating that SPD does improve the selection pressure of problem (4).

The above analysis validates that the set-based Pareto dominance relation proposed in this paper can well distinguish different set-based individuals, and can maintain a high selection pressure compared with the state-of-the-art dominance relations in the context of set-based evolution.

2) Performances of SetGA, IP-MOEA, and NSGA-IIIR: The difference between SetGA and two existing methods was first inspected by comparing the values of H, I, S and IGD metrics of the final fronts obtained by them, and the significance of the difference was further manifested by a non-parameter test, the Wilcoxon matched-pairs signed-rank test. Then, two methods were compared on T metric.

The means of H, I, S and IGD metrics obtained by IP-MOEA, NSGA-IIIR and SetGA were listed in Tables II-V, respectively. The superscript of a datum represents that this datum is significantly superior to that obtained by the corresponding method at a level of significance $\alpha = 0.01$, where the result was derived by the statistical software package SPSS V.17.0. More specifically, the datum whose superscript is S is significant superior to the corresponding datum obtained by SetGA, and the datum whose superscripts are I and N is significant superior to the corresponding data obtained by IPMOEA and NSGA-IIIR.

Tables II-V report that:

(1) For 40% of the problems, H metrics obtained by IP-MOEA are significantly superior to those obtained by SetGA; for 50% problems, SetGA is superior to IP-MOEA on H metric; for the remaining problems, there is no significant difference between them. For 28.2% of the problems, H metrics obtained by NSGA-IIIR are significantly superior to those obtained by SetGA; for 53.8% problems, SetGA is superior to NSGA-IIIR on H metric; for the remaining problems, there is no significant difference between them.

(2) All I metrics obtained by SetGA are significantly superior to those obtained by IP-MOEA. Furthermore, those obtained by SetGA are at least twice lower than those obtained by IP-MOEA. All I metrics obtained by SetGA are significantly superior to those obtained by IP-MOEA except for 5-objective DTLZ$_I$1 and DTLZ$_I$3, as well as 15-objective DTLZ$_I$1.

(3) For 92.5% of the problems, SetGA is significantly superior to IP-MOEA on S metric, and For 79.5% of the problems, SetGA is significantly superior to NSGA-IIIR on S metric. Nevertheless, for those problems, whose S metrics obtained by IP-MOEA or NSGA-IIIR are superior to those obtained by SetGA, the reason for the above circumstance may be that their bad convergence give rise to their good spread.

(4) For 76.9% of the problems, SetGA is significantly superior to IP-MOEA on IGD, and For 79.5% of the problems, SetGA is significantly superior to NSGA-IIIR on IGD.

It can be derived that for some optimization problems, the reason of SetGA superior to IP-MOEA and NSGA-IIIR on H metric is that hyper-volume and imprecision partly conflict with each other, which means that the decrease of imprecision does not always lead to the decrease of hyper-volume; in this case, both hyper-volume and imprecision can be improved, and they can simultaneously achieve the best. Whereas, also since hyper-volume and imprecision partly conflict with each other, for other optimization problems,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

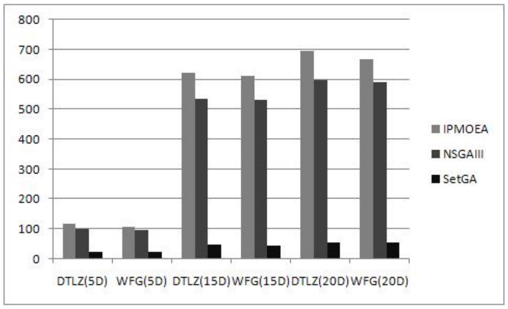

Fig. 6. CPU time of different methods

SetGA is inferior to IP-MOEA and NSGA-IIIR. In this case, optimizing imprecision may deteriorate hyper-volume, which results in its bad performance on H metric. Therefore, IP-MOEA and NSGA-IIIR, which take no account of uncertainty, are certainly more outstanding than SetGA, which are at the cost of uncertainty. As Bader *et al.* [30] revealed, if separately optimizing hyper-volume, SetGA will have a better convergence than IP-MOEA and NSGA-IIIR. Additionally, as imprecision is explicitly optimized and spread is the second selection criterion in this paper, SetGA automatically has a competitive performance on I and S metrics. For some cases, i.e., 5-objective DTLZ$_I$1 and DTLZ$_I$3 as well as 15-objective DTLZ$_I$1, the possible reason for NSGA-IIIR to outperform SetGA is that the objective optimized by NSGA-IIIR is the midpoint, which can be regarded as the sum of the lower limit and the radius; moreover, the radius represents the uncertainty of an interval. As a result, it is natural that NSGA-IIIR has a better performance than SetGA in a few cases. The superior performance of SetGA on IGD metric verifies its feasibility and effectiveness.

The above experimental analysis demonstrates that the dimensionality reduction based on hyper-volume and imprecision proposed in this paper and the sorting method for the set-based individuals with the same rank based on spread can produce a trade-off Pareto-optimal solution set between convergence and uncertainty, and that the obtained Pareto-optimal solution set has a good spread.

Fig. 5 illustrates average CPU time when optimizing different dimensional optimization problems by using three methods. It can be concluded from Fig. 5 that (1) the CPU time of SetGA for each type of optimization problems is significantly less than those of IP-MOEA and NSGAIIIR; (2) CPU time of the two methods increases along with the number of objectives, and the differences between IP-MOEA and SetGA as well as that between NSGAIIIR and SetGA increase along with the number of objectives and the population size, suggesting that SetGA is apparently better than IP-MOEA and NSGAIIIR on T metric, which supports the conclusion in subsection IV.C.

The above experimental results and analysis demonstrate that: (1) the proposed set-based Pareto dominance relation can significantly improve the selection pressure; as a result, the difficulty of traditional Pareto dominance relations in losing the selection pressure along with the number of objectives is solved; (2) SetGA proposed in this paper can reach a trade-off approximate front between convergence and uncertainty, accordingly, it balances optimality and stability of a system; (3) the evolution based on solution sets are superior to that based on solutions in the context of interval many-objective optimization, so it offers an effective candidate for solving IMaOPs.

VI. Conclusions

IMaOPs are ubiquitous. However, few effective methods for solving them exist as a result of their intrinsic complexity.

A SetGA for solving this kind of problems was presented in this paper. In this method, the original optimization problem was transformed into a deterministic bi-objective problem, where new objectives are hyper-volume and imprecision. In order to solve the converted problem, the proposed set-based Pareto dominance relation was employed to modify the non-dominated sorting approach in NSGA-II. Therefore, the selection pressure of the converted problem is increased. Additionally, several schemes for set-based evolution were proposed to produce a trade-off approximate front between convergence and uncertainty.

The complexity analysis of the algorithm and plentiful experimental results demonstrated that SetGA proposed in this paper can efficiently solve IMaOPs.

It is worth noting that the converted problem will be evidently different from problem (4), if other indictors are chosen as new objectives in the context of many-objective optimization. Besides, high-performance evolutionary schemes for solving the converted problem are beneficial to improving the performance of the final front. Therefore, constructing a novel converted optimization model and developing evolutionary schemes tailed to sets are our future research topics.

References

[1] E. J. Hughes, “Evolutionary multi-objective ranking with uncertainty and Noise,” in *Proceedings of 1st International Conference on Evolutionary Multi-Criterion Optimization*, Zurich, Switzerland, 2001, 329-343.

[2] M. Basseur, and E. Zitzler, “A preliminary study on handling uncertainty in indicator-based multiobjective optimization,” in *Proceedings of EvoWorkshops 2006*, Budapest, Hungary, 2006, 329-343.

[3] A. Syberfeldt, A. Ng, R. I. John, and P. More, “Evolutionary optimsation of noisy multi-objective problems using confident-based dynamic resampling,” *European Journal of Operational Research*, vol. 204, no. 3, pp. 533-544, Aug. 2010.

[4] V. A. Shim, K. C. Tan, J. Y. Chia, and A. A. Mamum, “Multi-objective optimization with estimation of distribution algorithm in a noisy environment,” *Evolutionary Computation*, vol. 21, no. 1, pp. 149-177, Jan. 2013.

[5] H. Karshenas, C. Bielza, and P. Larrañaga, “Interval-based ranking in noisy evolutionary multi-objective optimization,” *Computational Optimization and Applications*, vol. 61, no. 2, pp. 517-555, June 2015.

[6] D. W. Gong, N. N. Qin, and X. Y. Sun, “Evolutionary algorithms for optimization problems with uncertainties and hybrid indices,” *Information Sciences*, vol. 181, no. 19, pp. 4124-4138, Oct. 2011.

[7] S. T. Liu, “Using geometric programming to profit maximization with interval coefficients and quantity discount,” *Applied Mathematics and Computation*, vol. 209, no. 2, pp. 259-265, Mar. 2009.

[8] L. Majumder, and S. S. Rao, “Interval-based optimization of aircraft wings under landing loads,” *Computers and Structures*, vol. 87, no. 3-4, pp. 225-235, Feb. 2009.

[9] Z. H. Zhao, X. Han, C. Jiang, and X. X. Zhou, “A nonlinear interval-based optimization method with local-densifying approximation technique, ” *Structure Multidisplinary Optimization*, vol. 42, no. 4, pp. 559-573, Sep. 2010.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2016.2634625, IEEE Transactions on Evolutionary Computation

TABLE II
H METRIC OBTAINED BY DIFFERENT METHODS

	5D			15D			20D		
	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA
DTLZ$_I$1	1.000	1.000	0.9981	1.000	1.000	0.9916	1.0000	0.9999	0.9985
DTLZ$_I$2	0.9917^S	0.9942^S	0.8703	0.9825^S	0.9510^S	0.8954	0.9734^S	0.6692	0.8944^N
DTLZ$_I$3	1.000^S	1.000^S	0.9759	0.9896^S	0.9731^S	0.8699	0.9886^S	0.9375^S	0.8449
DTLZ$_I$4	0.9927^S	0.9954^S	0.9184	0.9909	0.6442	0.9994^N	1.0000^S	0.9438	0.9574
DTLZ$_I$5	0.1164	0.0227	0.1380^{I,N}	0.2805	0	0.9460^{I,N}	0.2464	0	0.9840^{I,N}
DTLZ$_I$6	0.4624^S	0.2247	0.2185	0.2643	0	0.5653^{I,N}	0.4068^S	0	0.1880
DTLZ$_I$7	0.9935^S	1.0305^S	0.5782	0.1334	0	0.2991^{I,N}	0.0284	0	0.0542^{I,N}
WFG$_I$1	0.1476	1.0600^S	0.1645^I	0.9328^S	2.3551^S	0.6532	0.0658^S	0.2305^S	0.0137
WFG$_I$2	0.9332	0.4930	1.0000^{I,N}	0.1340	0.0005	0.4690^{I,N}	0.0074	0.0019	0.0124^{I,N}
WFG$_I$3	0.9552	0.6234	1.0000^{I,N}	0.2004	0.0687	1.0000^{I,N}	0.0121	0.0960	0.0939^I
WFG$_I$4	0.4096	0.4400	1.0000^{I,N}	0.0616	0.0191	0.0687^{I,N}	0.0057	0.0061	0.3360^{I,N}
WFG$_I$5	0.9788	0.4458	1.0000^{I,N}	0.0662^S	0.0188	0.0551^N	0.0702	0.0907	0.1423^{I,N}
WFG$_I$6	0.9653^S	0.1943^S	0.1760	0.0079	0.0015	0.0390^{I,N}	0.0078	0.0031	0.0630^{I,N}
Car$_I$(9D)	0.9460^S		0.4290						

TABLE III
I METRIC OBTAINED BY DIFFERENT METHODS

	5D			15D			20D		
	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA
DTLZ$_I$1	76.856	1.0165^S	12.266^I	376.4	12.4990^S	37.42^I	260.35	80.465	38.455^{I,N}
DTLZ$_I$2	852.85	693.86	25.3709^{I,N}	698.81	561.69	76.976^{I,N}	6025.7	1774.8	78.073^{I,N}
DTLZ$_I$3	146.31	2.7740^S	20.8303^I	740.0	438.41	38.216^{I,N}	10557	1680	83.143^{I,N}
DTLZ$_I$4	1057.7	613.10	15.0003^{I,N}	7117	623.06	43.274^{I,N}	8778.8	1710.5	49.518^{I,N}
DTLZ$_I$5	286.62	286.17	59.753^{I,N}	312.36	8517.4	41.944^{I,N}	649.45	36426	78.518^{I,N}
DTLZ$_I$6	5440.0	2509.71	59.318^{I,N}	3571.2	8411.4	77.278^{I,N}	2073.6	30254	258.44^{I,N}
DTLZ$_I$7	390.9	615.07	59.679^{I,N}	2843	5797.4	111.13^{I,N}	6150	22373	22.282^{I,N}
WFG$_I$1	5340.6	2919.3	45.841^{I,N}	6367	2712.2	30.651^{I,N}	7731	15030	472.853^{I,N}
WFG$_I$2	13.132	1656.3	0.1255^{I,N}	4270	5782.5	11.469^{I,N}	5446	23823	568.26^{I,N}
WFG$_I$3	12.857	1831.5	0.0154^{I,N}	3532	3901.6	0.3246^{I,N}	4834	30577	448.96^{I,N}
WFG$_I$4	4409.3	2005.3	0.0272^{I,N}	51681	6916.4	232.52^{I,N}	3983	36333	8.2410^{I,N}
WFG$_I$5	102.30	2044.88	0.0847^{I,N}	2710	5333.6	288.07^{I,N}	4050	228.88	8.6862^{I,N}
WFG$_I$6	155.94	2350.31	38.6412^{I,N}	1993	6710.63	270.08^{I,N}	2470.5	31800	307.92^{I,N}
Car$_I$(9D)	24.8640		10.3607^I						

TABLE IV
S METRIC OBTAINED BY DIFFERENT METHODS

	5D			15D			20D		
	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA
DTLZ$_I$1	0.0860	0.0038	2.0491^{I,N}	0.0541	0.2169	3.0594^{I,N}	0.0423	0.3018	3.4746^{I,N}
DTLZ$_I$2	1.3648	0.9951	2.2257^{I,N}	1.0287	0.8829	3.5058^{I,N}	0.9612	1.0228	3.8002^{I,N}
DTLZ$_I$3	0.4336	0.1053	2.2201^{I,N}	1.1230	1.1968	3.4709^{I,N}	1.5428	1.3282	3.7476^{I,N}
DTLZ$_I$4	1.0319	0.9656	2.2278^{I,N}	1.0838	1.1420	3.8561^{I,N}	1.7177	1.3532	4.1455^{I,N}
DTLZ$_I$5	1.3722	1.1164	2.2260^{I,N}	2.1611	12.189^S	2.8983^I	3.2160^S	27.8071^S	2.8983
DTLZ$_I$6	2.0284	2.2437	2.2310^I	1.7900	1.3199	3.7112^{I,N}	3.3165^S	1.4885^S	1.3296
DTLZ$_I$7	0.5511	1.3698	2.2291^{I,N}	0.6848	2.3316	4.0800^{I,N}	1.0225	4.0605^S	3.3476^I
WFG$_I$1	1.8715	5.6127	11.2499^{I,N}	1.9232	5.6301	10.289^{I,N}	1.94184	7.4412	101.32^{I,N}
WFG$_I$2	1.3516	2.3152	4.1246^{I,N}	1.3516	2.6571	2.4467^I	1.5567	4.0457	101.60^{I,N}
WFG$_I$3	1.2876	2.3965	4.2364^{I,N}	1.2876^S	3.9487^S	1.2173	1.6253	0.2044	101.66^{I,N}
WFG$_I$4	2.2006	2.1796	6.8821^{I,N}	1.9499	4.7110	67.279^{I,N}	1.4912	9.0459^S	1.9970^I
WFG$_I$5	1.4576	2.296	3.9499^{I,N}	1.4576	3.9580	6.1471^{I,N}	1.4678	5.6505	101.72^{I,N}
WFG$_I$6	1.6433	2.3205	21.0642^{I,N}	1.6433	3.8960	67.3005^{I,N}	1.3499	5.2281	101.52^{I,N}
Car$_I$(9D)	1.7390		44.0034^I						

TABLE V
IGD OBTAINED BY DIFFERENT METHODS

	5D			15D			20D		
	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA	IP-MOEA	NSGA-IIIR	SetGA
$DTLZ_I1$	0.6328	0.6658	$0.2684^{I,N}$	0.5395^{S}	0.5705^{S}	0.8756	0.4836^{S}	0.5304^{S}	1.1546
$DTLZ_I2$	1.4483	1.3323	$0.3351^{I,N}$	2.0517	1.3170	$1.0394^{I,N}$	1.9866	1.6253	$1.4186^{I,N}$
$DTLZ_I3$	0.5534	0.6647	0.4001^{N}	0.8870^{S}	0.5504^{S}	1.0558	0.9743^{S}	0.9550^{S}	1.3170
$DTLZ_I4$	1.8607	1.3492	$0.3184^{I,N}$	1.8759	2.1664	$1.1119^{I,N}$	2.2338	2.1421	$1.4627^{I,N}$
$DTLZ_I5$	1.1115	1.5602	$0.3004^{I,N}$	38.1102	37.8619	$1.9381^{I,N}$	41.0298	51.6512	$2.6394^{I,N}$
$DTLZ_I6$	1.0409	1.5135	$0.2922^{I,N}$	40.2217	38.7148	$19.4774^{I,N}$	54.4987	52.0673	$26.5935^{I,N}$
$DTLZ_I7$	4.8993^{S}	5.2835^{S}	9.1835	39.2054	37.5187	$19.1835^{I,N}$	53.0248	51.5812	$26.1136^{I,N}$
WFG_I1	5.5747	2.9960^{S}	5.0871	6.3088	5.9132	5.1035^{I}	9.1835	8.2368	$6.2093^{I,N}$
WFG_I2	11.5791	11.4156	$5.8151^{I,N}$	37.8844	38.0101	$19.190^{I,N}$	53.7955	53.1117	$27.067^{I,N}$
WFG_I3	8.7560	11.3631	$5.7465^{I,N}$	40.7298	41.1652	$2.0786^{I,N}$	58.7420	59.7543	$29.3729^{I,N}$
WFG_I4	10.7939	11.0894	$5.2216^{I,N}$	41.593	41.030	$20.520^{I,N}$	59.759	59.6363	$29.7997^{I,N}$
WFG_I5	2.0080	2.23623	2.3297	5.9756	7.0557	5.1376^{N}	8.0119	8.8625	$6.4161^{I,N}$
WFG_I6	9.0184	10.8373	$5.4576^{I,N}$	37.3349	37.2762	$19.4291^{I,N}$	53.0944	53.5782	$27.2997^{I,N}$

[10] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength pareto evolutionary algorithm,” in *Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federa INstitute of Technology (ETH)*, Zurich, 2001.

[11] K. Deb, A. Pratap, S. Agarwal, and T. Mrystivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, Apr. 2002.

[12] E. Zitzler, and S. Kunzli, “Indicator-based selection in multiobjective search,” in *Proceedings of Parallel Problem Solving from Nature, PPSN VIII*, Birmingham, UK, 2004, 832-842.

[13] Q. F. 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, Dec. 2007.

[14] K. Deb, and H. Jain, “An evolutionary many-obective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving problems with box constraints,” *IEEE Transactions on Evolutionary Computation*, vol.18, no. 4, pp. 577-601, Aug. 2014.

[15] L. Sahoo, A. K. Bhunia, and P. K. Kapur, “Genetic algorithm based multi-objective reliability optimization in interval environment”, *Computers & Industrial Engineering*, vol.62, no. 1, pp. 152-160, Feb. 2012.

[16] A. K. Bhunia, and S. S. Samanta, “A study of interval metric and its application in multi-objective optimization with interval objectives”, *Computers & Industrial Engineering*, vol.74, pp. 169-178, Aug. 2014.

[17] P. Limbourg, and D. Aponte, “An optimizaiton algorithm for imprecise multi-objective problem function,” in *Proceedings of IEEE Congress on Evolutionary Computation*, Edinburgh, UK, 2005, 459-466.

[18] D. W. Gong, N. N. Qin, and X. Y. Sun, “Evolutionary optimization algorithm for multi-objective optimization problems with interval parameters,” in *Proceedings of 5th International Conference on Bio-Inspired Computing: Theories and Applications*, Changsha, China, 2010, 411-420.

[19] J. Sun, and D. W. Gong, “Solving interval multi-objective optimization problems using evolutionary algorithms with lower limit of possibility degree,” *Chinese Journal of Electronics*, vol.22, no. 2, pp. 269-272, Apr. 2013.

[20] S. X. Yang, M. Q. Li, X. H. Liu and J. H. Zheng, “A grid-based evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721-736, Oct. 2013.

[21] Y. Yuan, H. Xu, B. Wang and X. Yao, “A new domiance relation-based evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16-37, Feb. 2016.

[22] X. Y. Zhang, Y. Tian and Y. C. Jin, “Approximate non-dominated sorting for evolutionary many-objective optimization ,” *Information Sciences*, vol. 369, no. 10, pp. 14-33, Oct. 2016.

[23] Y. Xiang, Y. R. Zhou, M. Q. Li and Z. F. Chen, “A vector angle based evolutionary algorithm for unconstrained many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, DOI 10.1109/TEVC.2016.2587808.

[24] X. Y. Cai, Z. X. Yang, Z. Fan and Q. F. Zhang, “Decomposition-based-sorting and angle-based selection for evolutionary multiobjective and many-objective optimization,” *IEEE Transactions on Cybernetics*, 10.1109/TCYB.2016.2586191.

[25] H. K. Singh , A. Isaacs and T. Ray, “A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 539-554, Aug. 2011.

[26] D. K. Saxena, J. A. Duro, A. Tiwari, K. Deb and Q. F. Zhang, “Objective reduction in many-objective optimization: Linear and nonlinear algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 1, pp. 77-99, Feb. 2013.

[27] S. Bandyopadhyay, and A. Mukherjee, “An algorithm for many-objective optimization with reduced objective computations: A study in differential evolution,” *IEEE Transactions on Evolutionary Computation*, vol.19, no. 3, pp. 400-413, June 2015.

[28] Z. N. He and G. G. Yen, “Many-objective evolutionary algorithm: Objective space reduction and diversity improvement,” *IEEE Transactions on Evolutionary Computation*, vol.20, no. 1, pp. 145-160, Feb. 2016.

[29] Y. M. Cheung, F. Q. Gu and H. L. Liu, “Objective extraction for many-objective optimization problems: Algorithm and test problems,” *IEEE Transactions on Evolutionary Computation*, DOI 10.1109/TEVC.2016.2519758.

[30] J. Bader, D. Brockhoff, S. Welten, and E. Zitzler, “On using populations of sets in multi-objective optimization,” in *Proceedings of 5th International Conference on Evolutionary Multi-Criterion Optimization*, Nantes, France, 2009, 140-154.

[31] E. Zitzler, L. Thiele, and J. Bader, “On set-based multiobjective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, pp. 58-79, Jan. 2010.

[32] J. Bader, and E. Zitzler, “HypE: An algorithm for fast hyper-volume-based many-objective optimization,” *Evolutionary Computation*, vol. 19, no. 1, pp. 45-76, Jan. 2011.

[33] R. E. Moore, R. B. Kearfott and M. J. Cloud, “Introduction to Interval Analysis,” SIAM: Philadelphia, 2009.

[34] R. Berghammer, T. Friedrich, and F. Neumann, “Set-based multi-objective optimization, indicators, and deteriorative cycles,” in *Proceedings of 12th Annual Genetic and Evolutionary Computation Conference*, Portland, USA, 2010, 1-8.

[35] G. Rudolph, “On geometrically fast convergence to optimal dominated hyper-volume of set-based multiobjective evolutionary algorithms,” in *Proceedings of the IEEE Congress on Evolutionary Computation*, New Orleans, USA, 2011, 1719-1723.

[36] X. Y. Sun, R. D. Xu, Y. Zhang, and D. W. Gong, “Sets evolution-based particle swarm optimization for many-objective problems,” in *Proceedings of the IEEE International Conference on Information and Automation*, Hailar, China, 2014, 1119-1124.

[37] J. Knowles, and D. Corne, “On metrics for comparing nondominated sets,” in *Proceedings of 2002 World Congress on Computational Intelligence*, Honolulu, USA, 2002, 711-716.

[38] N. Beume, B. Naujoks, and M. Emmerich, “SMS-EMOA:Multiobjective selection based on dominated hyper-volume,” *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653-1669, Sep. 2007.

[39] E. Zitzler, “Evolutionary algorithms for multiobjective optimization: methods and applications,” [Ph. D. Dissertation], Swiss Federal Institute of Technology, 1999.

[40] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable multi-objective optimization test problems,” in *Proceedings of 2002 World Congress on Computational Intelligence*, Honolulu, USA, 2002, 825-830.

[41] S. Huband, L. Barone, L. While, Phil Hingston, “A scalable multi-objective test problem toolkit,” in *Proceedings of 3rd International*

Conference on Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico, 2005, 280-295.
[42] H. Jain, and K. Deb, “An improved NSGA-II procedure for many-objective optimization Part I: solving problems with box constraints,” in *KanGAL Report Number 2012009*, Department of Mechanical Engineering Indian Institute of Technology, Kanpur.
[43] https://github.com/O-T-L

Dunwei Gong received the B.S. degree in applied mathematics from China University of Mining and Technology, Xuzhou, P.R.China, in 1992, M.S. degree in control theory and its applications from Beijing University of Aeronautics and Astronautics, Beijing, P.R.China, in 1995, and Ph.D. in control theory and control engineering from China University of Mining and Technology, Xuzhou, P.R.China, in 1999. Since 2004, he has been a professor in School of Information and Electrical Engineering, China University of Mining and Technology. He takes up the director of the Institute of Automation, China University of Mining and Technology. His main research interests are evolutionary computation and search based software engineering.

Jing Sun received the B.S. degree in mathematics education from Center China Normal University, Wuhan, P.R.China, in 1997, M.S. degree in applied mathematics from China University of Mining and Technology, Xuzhou, P.R.China, in 2003, and Ph.D. in control theory and control engineering from China University of Mining and Technology, Xuzhou, P.R.China, in 2012. Since 2010, she has been an associate professor in College of sciences, Huaihai Institute of Technology. Her main research interests are evolutionary computation, interval multi-objective optimization, and machine learning.

Zhuang Miao received the B.S. degree in electrical engineering and automation from China University of Mining and Technology, Xuzhou, P.R.China, in 2014. He is now the M.S. degree candidate in control theory and control engineering of China University of Mining and Technology, Xuzhou, P. R. China. His main research interests are evolutionary computation and interval optimization.