Online Diversity Assessment in Evolutionary Multiobjective Optimization: A Geometrical Perspective

Sen Bong Gee, Kay Chen Tan, *Fellow, IEEE,* Vui Ann Shim, and Nikhil R. Pal, *Fellow, IEEE*

***Abstract*—Many diversity metrics have been proposed for offline diversity measurement of the whole population in multiobjective optimization. Most of the existing methods require knowledge of the exact Pareto optimal front or the ideal vector. For this reason, there is no direct approach to use the diversity metrics in an online manner. In this paper we propose an online diversity metric that is inspired by the geometrical interpretation of convergence and diversity. In addition, the proposed method is able to measure the diversity loss caused by any individual in the population. This information is useful in the selection process as the algorithm can perform a diversity-preservation selection based on the measured diversity loss contributed by each individual. To demonstrate the effectiveness of the proposed metric in enhancing the diversification of the solution set, we implement the metric on the well-known multiobjective evolutionary algorithm with decomposition. The simulation results show the applicability and usability of the proposed online diversity measurement.**

***Index Terms*—Diversity assessment, evolutionary algorithm, multiobjective.**

I. Introduction

MULTIOBJECTIVE optimization involves more than one objective function in the process of optimization, leading to the notion of optimum that is intrinsically different from the notion of optimum in single-objective optimization problems (SOPs). In multiobjective optimization problems (MOPs), the main task of any evolutionary multiobjective optimization (EMO) algorithm is to find the best set of trade-off solutions between different objective functions. Generally, the objective functions are conflicting to each other, which results in a multitude of trade-off solutions in the objective space. Since it is impractical to find all trade-off solutions, the task of the EMO algorithm is to approximate the Pareto optimal front (POF) that is formed by the best trade-off solutions in the objective space [1]–[3].

Manuscript received December 13, 2013; revised April 19, 2014 and June 24, 2014; accepted July 21, 2014. Date of publication September 17, 2014; date of current version July 28, 2015. This work was supported by the Singapore Ministry of Education Academic Research Fund Tier 1 under Project R-263-000-A12-112.

S. B. Gee and K. C. Tan are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: a0039834@nus.edu.sg; eletankc@nus.edu.sg).

V. A. Shim is with the Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore 138632 (e-mail: shimva@i2r.a-star.edu.sg).

N. R. Pal is with the Electronics and Communication Sciences Unit, Indian Statistical Institute, Calcutta 700108, India (e-mail: nikhil@isical.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2014.2353672

Diversity and convergence are two critical issues of a search process in any EMO algorithm. Given an EMO algorithm, the closeness of the output solutions to POF is related to the convergence properties of the algorithm whereas the spread and distribution of the output solutions are related to the diversity properties. Convergence and diversity of an algorithm are often conflicting goals as strong convergence properties most likely result in poor diversity of solutions and vice versa. The balance between these two properties is important to the design of an EMO algorithm [4].

In order to determine the optimization performance of an EMO algorithm, diversity metric is often used to estimate the diversity of the generated solution set along the POF. This diversity metric is significantly different from the diversity metric used in SOP as the latter measures diversity of whole population without considering the POF. For the ease under discussion, we use "multiobjective diversity metric" to denote the former whereas "single-objective diversity metric" to denote the latter. The notion of multiobjective diversity is related to the spread and distribution of output solutions along the POF. Numerous ways of measuring multiobjective diversity based on these two perspectives have been proposed and used to assess the performance of EMO algorithms [5]–[22].

Another interesting application of a diversity metric is to assess the diversity of solutions in an online manner and use the metric to monitor and guide the search process [15], [16], [23]. Comparing to single-objective optimization, there are relatively few studies on online solution diversity metrics to guide the search by means of operator adaption. One of the possible reasons is that there is no direct way to convert the offline multiobjective diversity metrics into online diversity metrics as majority of the offline metrics require the knowledge of POF. Recently, there is a trend to use hypervolume indicator (S-metric or Lebesgue measure, first introduced into MOP by Zitzler and Thiele [24], more detailed treatments can be referred to [10], [17], and [25]–[36]) to guide the search as it does not require POF to be known *a priori*. Hypervolume indicator includes the convergence and diversity information in a single metric. Since this metric contains more than one type of information, there is no way to separate the convergence and diversity information from a single metric. Knowing the convergence and diversity related information individually would definitely benefit design flexibility of the EMO algorithm, especially for adaptive EMO algorithms.

The knowledge of the solution diversity in the objective space can be beneficial to the design of selection and genetic operators. One popular utilization of this information is to avoid selecting similar solutions in the objective space. For instance, Deb *et al.* [14] used crowding distance in nondominated sorting genetic algorithm II (NSGA-II) to select the less crowded nondominated solutions. Zitzler and Thiele [37] introduced strength Pareto evolutionary algorithm (SPEA), which applies the clustering technique in selection operator to preserve the solution diversity. Moreover, there are numerous EMO algorithms that use grid [38]–[42] or entropy [12], [43]–[45] to estimate the solution diversity. These algorithms explicitly utilize the knowledge of solution diversity to maintain the population diversity hoping that it can maintain the search ability of the algorithm. Another use of the solution diversity is to enhance the exploitation properties of an algorithm when the diversity of solutions is low. A pertinent and quantitative description of the solution diversity is required for the control of exploitation properties of the algorithm. Similar concepts can be found in [46] and [47] where they utilize the information of solution diversity to control the exploitation and exploration properties of the algorithm. Although the concept is similar, there are some crucial differences. In Silva and Burke's method [46], single-objective diversity metric is used instead of multiobjective diversity metric. Furthermore, Chow and Yuen [47] use the solution diversity in decision space as opposed to in the objective space.

To the best of the authors' knowledge, there are few studies that focus on online quantitative description of the multiobjective solution diversity in the objective space. There remains a need for an online diversity metric that can be used in the algorithm design. In this paper, we attempt to fill this gap and introduce an online diversity metric to estimate diversity among the nondominated set. This paper also extends our previous work in [48] on diversity metric.

II. Background

In this section, we go through some basic definitions used in the EMO community and present some early works in diversity measurement.

A. Some Basic Definitions

A MOP consists of more than one objective function that needs to be optimized simultaneously. Mathematically, an MOP can be expressed as

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}} \quad \mathbf{f}(\mathbf{x}) = \left[f_1(\mathbf{x})\ f_2(\mathbf{x})\ \ldots\ f_m(\mathbf{x})\right]^T \\ &\text{subject to} \quad \mathbf{x} \in \Omega \end{aligned} \tag{1}$$

where f_i is the ith objective function; m is the total number of objective functions; $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m$ is the objective vector; n is the total number of decision variables; $\mathbf{x} \in \mathbb{R}^n$ is the decision vector and $\Omega \subset \mathbb{R}^n$ is the feasible decision space. Without loss of generality, a minimization problem is considered here. Generally, Ω can be described by

$$\begin{aligned} \Omega = \big\{\mathbf{x} \in \mathbb{R}^n | g_j(\mathbf{x}) \le 0 \text{ for } j = 1, \ldots, p; \\ h_k(\mathbf{x}) = 0, \text{ for } k = 1, \ldots, q\big\} \end{aligned} \tag{2}$$

where g_j is the jth inequality constraint out of total p inequality equations and h_k is the kth equality constraint out of total q equality equations. Since (1) involves more than one objective function, it might be no single point in Ω that minimizes all the objectives simultaneously. Therefore, the concepts of Pareto dominance and Pareto optimality are needed to define the solution set that provides the best trade-off between different objective functions.

Let $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$ be any two objective vectors. For the case of minimization, $\mathbf{u}$ is said to dominate $\mathbf{v}$ (or $\mathbf{u} \preceq \mathbf{v}$) if and only if $u_i \le v_i$ for all $i = 1, 2, \ldots, m$ and there exists at least one $u_i < v_i$. Let $F \subset \mathbb{R}^m$ be the feasible objective space that is mapped by Ω. A decision vector $\mathbf{x}^* \in \Omega$ is said to be a Pareto optimal solution if there is no vector in F that dominates $\mathbf{f}(\mathbf{x}^*)$. The objective vector of the Pareto optimal solution is called Pareto optimal objective vector. All the Pareto optimal solutions in Ω jointly form the Pareto optimal set (POS) whereas all the Pareto optimal objective vectors in F collectively form the Pareto optimal front (POF). A vector whose elements are the lower (upper for maximization problem) bounds for the objective function values of Pareto optimal solutions is called the ideal vector. In general, the ideal vector is in the infeasible objective space. For more detailed treatment on these concepts, please see [1]–[3].

B. Early Works in Diversity Measurement

Most of the early pioneering works on multiobjective diversity measurement focus on offline assessment of the approximated Pareto front quality.

Deb's chi-square-like deviation [5] is one of the earliest works on diversity measurement. This approach was inspired by the distribution measurement in statistics. Later, Schott [6] suggested spacing method in his thesis to evaluate the performance of a multiobjective evolutionary algorithm (MOEA). This metric is used to measure the distribution of the approximated Pareto solutions. As per [7], this performance metric may be misleading as low value of this metric does not necessarily imply the good distribution of the approximated Pareto solutions. Apart from focusing on solution distribution, Zitzler [8] suggested to use the spread of the objective space solutions to evaluate the algorithm performance. Tan *et al.* [9] proposed another metric, namely uniformly distribution index, which uses the standard deviation of the niche count to gauge the distribution of the solutions.

Deb *et al.* [11] proposed a diversity metric that can evaluate the spread and distribution of the solutions. One of the weaknesses of this performance metric is that it is restricted to two-objective problems. Farhang-Mehr and Azarm [12] proposed an entropy approach to measure the diversity of solutions. This entropy metric quantifies the information of the solutions and evaluates performance of an algorithm based on the total information of the solutions. Later, Deb and Jain [13] suggested grid diversity metric to measure the diversity of the solution. Deb *et al.* [14] defined sparsity measure to evaluate the distribution of solutions. Both metrics in [13] and [14] require a properly defined hyperplane to project the obtained solutions on it. The appropriateness of these two metrics

are sensitive to the choice of hyperplane and the inherent dimension of the POF.

There is no direct way to use any of the above-mentioned measurement methods in an online manner as they require the knowledge of the POF or the ideal vector. In the literature, there are relatively few studies on online multiobjective diversity metrics. Mostaghim and Teich [15], [16] proposed a sigma method in multiobjective particle swarm optimization (MOPSO) to guide the search process. This metric imposes a strict assumption that the POF lies in the positive objective space. Silva and Burke [46] used a single-objective diversity metric to monitor the exploration and exploitation balance of the EMO algorithm as this type of metric does not require the knowledge of POF to compute. Since single-objective optimization is intrinsically different from multiobjective optimization [49], single-objective diversity metric may not be able to capture the distribution and spread information required in multiobjective optimization. Tan *et al*. [44] proposed another technique to compute the entropy of the solutions by using Parzen window density estimation. This approach is sensitive to kernel width setting as mentioned in the paper.

Although previous research works have successfully used their diversity metrics in many tasks, there remains a need for a flexible and accurate online diversity metric that can be used in the algorithm design. In the following sections, a geometry-based online diversity metric is illustrated and examined.

III. Online Multiobjective Diversity Metric

In this section, we first investigate the geometrical implication of the convergence and diversity of any EMO algorithm. Then, we present an implementation to estimate the convergence direction during the search process. Based on the estimated convergence direction, the proposed diversity metric can be calculated in an online manner.

A. Geometrical Interpretation of the Diversity and Convergence

There is no consensus on the definition of convergence and diversity as it can be defined in various ways. To avoid confusion, we define what we consider to be convergence and diversity. Convergence is defined here as the notion that the population-solutions approach or approximate the POF in the objective space. On top of this definition, we extend the definition of convergence to convergence direction. Convergence direction is referred to as the direction that is used to describe the path of the solutions approaching the POF. As discussed in the introduction section, diversity is defined as the spread and distribution of solutions along the POF in the objective space. In this paper, we define diverse direction as the direction that is perpendicular to the convergence direction. Diverse direction is closely related to the distribution of solutions around the convergence direction. In Fig. 1, the dotted arrows denote the convergence direction whereas the double head arrows denote the diverse direction. Note that the convergence and diverse direction are defined such that they are perpendicular to each other.

In the view of the fact that majority of the existing offline multiobjective diversity metrics require the knowledge of POF or the ideal vector, this limits the applicability of these metrics to guide the search as the knowledge of POF or the ideal vector is not available during the search process. To circumvent the problem, we design a novel diversity metric that uses the knowledge of convergence direction instead of POF or the ideal vector. In the following subsections, estimation of the convergence direction and calculation of the online diversity metric will be discussed in detail.

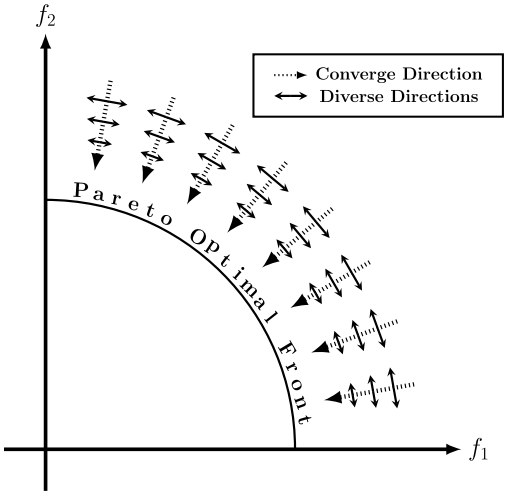

Fig. 1. Arbitrary drawn POF. The dotted arrows and double head arrows denote the convergence and diversity-related directions.

B. Estimation of the Convergence Direction

Although the exact convergence direction is not available during the search, it is possible to estimate the convergence direction. To estimate the convergence direction, we must extract the information from the parent and offspring population as they are the only resources that provide useful information during the search process. One simple way to estimate the convergence direction is to use one parent and one offspring solution, and the offspring solution must dominate the parent solution. In addition, the parent solution that is closest to the offspring solution (in the objective space) is selected to be the candidate for the estimation of convergence direction. The estimated convergence direction is mathematically defined as

$$\mathbf{d}_{\text{conv},i} = \mathbf{c}_j - \mathbf{p}_i \tag{3}$$

where $\mathbf{p}$ and $\mathbf{c}$ are the parent and offspring objective vectors, j is from the index set of

$$D = \{d | \exists \mathbf{c}_d \prec \mathbf{p}_k, k \in [1, \ldots, N], d \in [1, \ldots, |C|]\}. \tag{4}$$

In (4), the corresponding offspring solution of the index d must weakly dominate at least one parent solution. For each index j in (3), we would like to find the corresponding parent solution $\mathbf{p}_i$ that is closest to $\mathbf{c}_j$ and $\mathbf{p}_i$ is weakly dominated by $\mathbf{c}_j$. Therefore, i is defined as

$$i = \underset{k \in D_p}{\operatorname{argmin}} \left|\left|\mathbf{p}_k - \mathbf{c}_j\right|\right| \tag{5}$$

where

$$D_p = \left\{k | \exists \mathbf{c}_j \prec \mathbf{p}_k, k \in [1, \ldots, N]\right\}. \tag{6}$$

N is the parent population size, and C denote the offspring population. D_p in (6) denotes the index set of parent solutions that are weakly dominated by $\mathbf{c}_j$. There are numerous ways to estimate the convergence direction. For instance, we can impose additional restriction on the number of times that a parent objective vector $\mathbf{p}$ that can be used in the computation of convergence direction. Throughout this paper, (3)–(6) are used to estimate the convergence direction due to its simplicity.

C. Calculation of the Online Diversity Metric

To compute the proposed online diversity metric, we need to use the information of convergence direction. We first define relative diversity loss (RDL), which is a diversity measurement quantity that indicates the amount of diversity loss of an individual solution between two consecutive generations with respect to a particular convergence direction. This quantity is basically a ratio between two lines: the shortest distances of parent and offspring solution to the line of convergence direction. Mathematically, this quantity is defined as

$$\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}} = \frac{\left|\left|\mathbf{p}' - \mathbf{proj}_{d_{\text{conv},i}}\mathbf{p}'\right|\right|}{\left|\left|\mathbf{c}' - \mathbf{proj}_{d_{\text{conv},i}}\mathbf{c}'\right|\right|} \tag{7}$$

where

$$\mathbf{p}' = \mathbf{p} - \mathbf{p}_r$$
$$\mathbf{c}' = \mathbf{c} - \mathbf{c}_r.$$

$\mathbf{p}_r$ and $\mathbf{c}_r$ are the parent and offspring objective vectors used to form the convergence direction in (3); $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$ denotes the RDL with respect to convergence direction $\mathbf{d}_{\text{conv},i}$; $\mathbf{proj}_{d_{\text{conv},i}}\mathbf{p}'$ and $\mathbf{proj}_{d_{\text{conv},i}}\mathbf{c}'$ are the projection of $\mathbf{p}'$ and $\mathbf{c}'$ objective vectors onto the convergence direction $\mathbf{d}_{\text{conv},i}$. The numerator of (7) denotes the closest distance between the parent solution ($\mathbf{p}$) to the convergence direction ($\mathbf{c}_r - \mathbf{p}_r$), whereas the denominator denotes the closest distance between the offspring solution ($\mathbf{c}$) to the convergence direction ($\mathbf{c}_r - \mathbf{p}_r$). If there are only two individuals in the population, this value can represent the estimation of the ratio between the spread of parent solutions and the spread of offspring solutions over two generations. High value of this ratio implies a significant reduction of the solution spread. This can be an indication of the amount of diversity loss over consecutive two generations.

To further illustrate the way of computing RDL, a simple example is given. Assume that there are a parent pair (A and B) and an offspring pair (C and D) solutions. We use $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ to represent the objective vectors of solutions A, B, C, and D, respectively. At the beginning of the generation, there is no convergence direction in the archive. Assume that offspring C is generated and evaluated before offspring D in Fig. 2. Since offspring C dominates both parents A and B, it will replace one of them. Based on the previous-mentioned estimation method, we can form a convergence direction $\mathbf{d}_{\text{conv},1} = \mathbf{c} - \mathbf{a}$ and put it into the convergence direction archive. As parent A is near to offspring C, offspring C will replace parent A. When offspring D is generated, we can check whether it dominates parent B or not. From the figure, it is clear that offspring D dominates parent B. Since the convergence archive is no longer

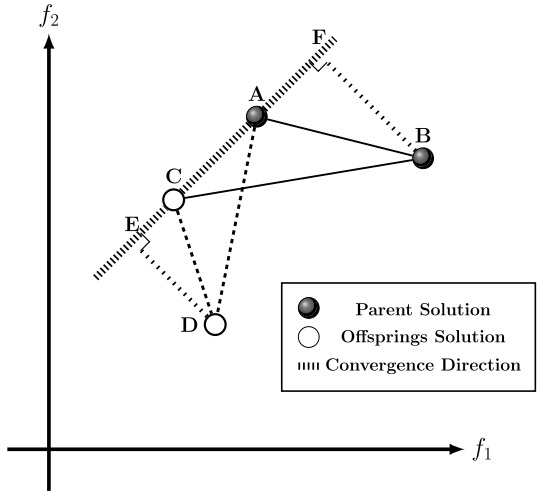

Fig. 2. Parent A and offspring C form an estimated convergence direction, which will be used to compute the relative diversity loss of parent B to offspring D.

empty, we can compute $\Gamma^{\mathbf{b}\to\mathbf{d}}_{\mathbf{d}_{\text{conv},1}}$. The rejections of parent $\mathbf{b}$ and offspring $\mathbf{d}$ from the first convergence direction ($\mathbf{d}_{\text{conv},1}$) are BF and DE, respectively. By basic trigonometry, the ratio between BF and DE can be computed as

$$\Gamma^{\mathbf{b}\to\mathbf{d}}_{\mathbf{d}_{\text{conv},1}} = \frac{\triangle ABC}{\triangle ADC} \tag{8}$$

where $\triangle ABC$ and $\triangle ADC$ are the areas of triangle ABC and ADC, respectively. We invoke Heron's formula to calculate the triangle area

$$\text{Area of } \triangle = \sqrt{s(s-a)(s-b)(s-c)} \tag{9}$$

where a, b, c are the sides of triangle and $s = (1/2)(a+b+c)$. If offspring D is first generated, similar calculation procedure can be used to obtain $\Gamma^{\mathbf{b}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},2}}$, where $\mathbf{d}_{\text{conv},2} = \mathbf{d} - \mathbf{a}$. For MOPs involving more than two objectives, a similar procedure is applied as (8) and (9) only involve the distances between different individuals in the objective space. In this case, three individuals form a triangle in the objective space. Heron's formula can still be used to compute the triangle based on the distances between any two individuals.

During the evolutionary search process, there are generally more than one convergence direction at any generation. To estimate the diversity loss of a solution to the whole population, we introduce the notion of maximum relative diversity loss (MRDL). Given k convergence directions, MRDL can be obtained by

$$\Gamma^{\mathbf{p}\to\mathbf{c}} = \max_{i=1,\ldots,k} \Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}} \tag{10}$$

where $\Gamma^{\mathbf{p}\to\mathbf{c}}$ is the MRDL. The magnitude of $\Gamma^{\mathbf{p}\to\mathbf{c}}$ is governed by the highest $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$ among the k convergence directions. Below are some properties and implications of the $\Gamma^{\mathbf{p}\to\mathbf{c}}$ and $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$.

1) $\Gamma^{\mathbf{p}\to\mathbf{c}}$ must be nonnegative, i.e., $\Gamma^{\mathbf{p}\to\mathbf{c}} \notin \mathbb{R}^-$.
2) If a newly generated offspring solution is identical to any offspring solution that is in the convergence archive, $\Gamma^{\mathbf{p}\to\mathbf{c}}$ will be infinite.

3) If a parent solution is identical to any parent solution that is in convergence archive, $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$ will be zero.
4) High value of $\Gamma^{\mathbf{p}\to\mathbf{c}}$ indicates either the existence of a similar offspring solution in the convergence archive or the offspring solution is close to the line of estimated convergence direction. If the offspring solution is exactly on the line, the $\Gamma^{\mathbf{p}\to\mathbf{c}}$ must be infinite. The similarity here is based on Euclidean distance of two vectors in the objective space.

Throughout the search process, the inherent nature of an EMO algorithm exerts selection pressure to the trade-off region. Making use of this property, we define a metric to estimate the reduction of spread of any particular solution between two consecutive generations. This metric is different from other existing ones as it measures the diversity loss caused by an individual solution.

IV. Algorithm Design

This section presents an implementation of the proposed online diversity metric on the well-known MOEA with decomposition (MOEA/D) [50], [51]. MOEA/D is an evolutionary algorithm that decomposes any given MOP into a number of single-objective subproblems. It optimizes each subproblem simultaneously during the evolutionary search process. To decompose the MOP, we use Tchebycheff approach [3] throughout this paper.

A. Improved Method of Generating Evenly Spread Vectors

When the number of objectives is large, there is no easy way to generate the evenly spread weight vectors in MOEA/D. The task of generating weight vectors is not trivial as it affects the spread of final solutions. Evenly spread weight vectors are typically preferable as we have no prior knowledge on how the weight vectors affect the distribution of output solutions. If we scrutinize the evenly spread weight vectors carefully, it is not hard to find that the task of generating evenly spread weight vectors is actually a recursive problem. To ease the task of generating evenly spread weight vectors, we propose a recursion method to generate evenly spread weight vectors as shown in Algorithm 1. In the pseudocode, i_j denotes the jth element in the vector i. Besides, $i_{1\to j}$ denotes the first element to the jth element in the vector i (1-based numbering is used for the vector index). To illustrate the recursive nature of generating the weight vector, a two-objective weight vector generation example is given as shown in Fig. 3.

To use the algorithm, the number of objective functions and the desired difference between elements in weight vectors are required to be specified. The minimum non-zero difference between elements in a weight vector is equal to $1/H$ where H is a natural number that controls the total number of weight vectors generated. The total number of generated weight vectors is

$$N = C^{m-1}_{H+m-1} \tag{11}$$

where C^k_r denotes the number of k-combinations from r elements.

Algorithm 1 Weight (*index*, *pre*)

Require:
m: Number of objective functions
interval: Difference between weight vectors
Initial Condition: $index = 1$; $pre_{1\to m} = 0$;
$index \in \mathbb{N}$; $interval \in \{1, 1/2, \ldots, 1/H\}$ where $H \in \mathbb{N}$;
$pre, output \in \mathbb{R}^m$;
$W = \emptyset$

Ensure: A set of evenly spread weight vectors (W)

1: **if** $index \leq m$ **then**
2: **if** sum($pre_{1\to index}$) > 1 **then**
3: **return**
4: **else**
5: copy *pre* to *output*
6: **if** $index = m$ **then**
7: $output_m = 1 -$ sum($pre_{1\to m}$)
8: $W = W \cup output$
9: **else**
10: copy *pre* to *temp*
11: weight($index + 1$, *temp*)
12: $pre_{index} = pre_{index} + interval$
13: weight(*index*, *pre*)
14: **end if**
15: **end if**
16: **else**
17: **return**
18: **end if**

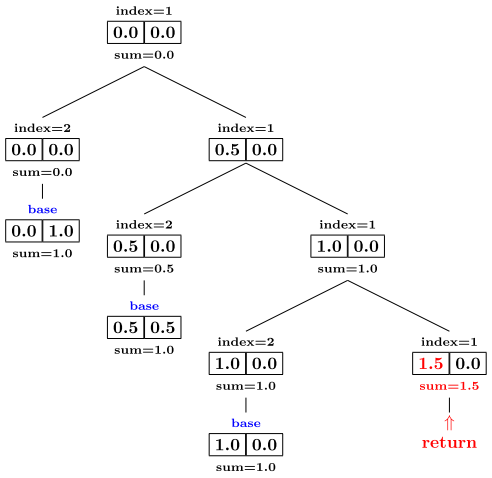

Fig. 3. Algorithm 1 with two objectives ($m = 2$). In this example, the control parameter H is 2 ($interval = 1/H = 1/2$). The "base" represents the base case, which corresponds to lines 6–8 in Algorithm 1. The node in the bottom right (where first value in the box equals to 1.5) corresponds to the lines 2–3 in Algorithm 1. The left and right nodes correspond to lines 11 and 13, respectively. When the weight generation process reaches the base case, the last element of the weight vector equals to unity subtracts the sum of previous elements in the weight vector (line 7).

B. Selection Operator

A new selection operator is implemented in MOEA/D to replace the original counterpart. In the new selection operator, each offspring solution is checked for its MRDL (or equivalently, $\Gamma^{\mathbf{p}\to\mathbf{c}}$) before the parent-offspring replacement happens. If the diversity loss of a given offspring solution

Algorithm 2 Pseudocode of Environmental Selection

Require:
P_i: Parent population at i-th generation.
C_i: Offspring population at i-th generation.
Condition: $|P_i| = |C_i|$
Ensure:
P_{i+1}: Parent population in the next generation.
Declare a set $D = \{1, \ldots, |P_i|\}$ and permute the sequence in the set. Set $P_{i+1} = P_{\text{conv}} = C_{\text{conv}} = \emptyset$.
for $k = 1$ to $|C_i|$ **do**
  Compare $C_i^{D(k)}$'s Tchebycheff scalar function with its neighboring parents. Store the neighboring parent solutions that has worse scalar function value than $C_i^{D(k)}$'s to P_i^M.
  if $P_i^M = \emptyset$ **then**
    $P_{i+1} = P_{i+1} \cup P_i^{D(k)}$
  else
    Find the nearest parent P_i^m from P_i^M(to $C_i^{D(k)}$).
    if $P_{\text{conv}} = C_{\text{conv}} = \emptyset$ **then**
      $P_{i+1} = P_{i+1} \cup C_i^{D(k)}$
      $P_{\text{conv}} = P_{\text{conv}} \cup P_i^m$
      $C_{\text{conv}} = C_{\text{conv}} \cup C_i^{D(k)}$
    else
      Compute $\Gamma^{\mathbf{p}\to\mathbf{c}}$ of each solution in P_i^M to $C_i^{D(k)}$ using Algorithm 3.
      if any of $\Gamma^{\mathbf{p}\to\mathbf{c}} > \gamma$ **then**
        $P_{i+1} = P_{i+1} \cup P_i^{D(k)}$
      else
        $P_{i+1} = P_{i+1} \cup C_i^{D(k)}$
        $P_{\text{conv}} = P_{\text{conv}} \cup P_i^m$
        $C_{\text{conv}} = C_{\text{conv}} \cup C_i^{D(k)}$
      end if
    end if
  end if
end for

Algorithm 3 Compute $\Gamma^{\mathbf{p}\to\mathbf{c}}$

Require:
P_{conv}: Parents objective vector set used in estimation of convergence direction.
C_{conv}: Offspring objective vector set used in estimation of convergence direction.
$\boldsymbol{x}$: The parent objective vector.
$\boldsymbol{y}$: The offspring objective vector.
Require: $|P_{conv}| = |C_{conv}|$
Ensure:
$\Gamma^{\mathbf{p}\to\mathbf{c}}$: Maximum Relative Diversity Loss
if $P_{\text{conv}} = \emptyset$ **then**
  $\Gamma^{\mathbf{p}\to\mathbf{c}} = 0$
  return
else
  $size = |P_{conv}|$
  $max = 0$
  for $i = 1 \to size$ **do**
    Calculate $r = \frac{\triangle \boldsymbol{x} P_{\text{conv},i} C_{\text{conv},i}}{\triangle \boldsymbol{y} P_{\text{conv},i} C_{\text{conv},i}}$ using equation (8)
    if $r > max$ **then**
      $max = r$
    end if
  end for
  $\Gamma^{\mathbf{p}\to\mathbf{c}} = max$
  return
end if

is higher than a predefined value (γ), the parent-offspring replacement is called off and the selection operator selects the parent solution instead of offspring solution. The rationale behind this mechanism is to maintain the diversity of solutions along the POF. Two archives are needed to store the convergence direction vector: one for the offspring solutions and another for the parent solutions. The pseudocode for the selection operator is shown in Algorithm 2 and the procedure to calculate $\Gamma^{\mathbf{p}\to\mathbf{c}}$ is shown in Algorithm 3.

The environmental selection in Algorithm 2 is to select the N individual solutions to survive in the next generation. In MOEA/D, each individual has a specific weight vector that is used to decompose the given MOP into several SOPs. The neighbors of each weight vector is stored in a set for each individual. During the selection, each offspring solution (of a specific weight vector) compares with neighboring parent solutions. If the offspring has a lower scalar function value (in this paper, we use Tchebycheff to decompose the MOP), the algorithm will compute $\Gamma^{\mathbf{p}\to\mathbf{c}}$ of each dominated parent to the offspring. If any of the $\Gamma^{\mathbf{p}\to\mathbf{c}}$ is greater than the predefined value (γ), the algorithm will select the parent solution, $P_i^{D(k)}$. Otherwise, the offspring solution, $C_i^{D(k)}$, will be selected to be the parent solution in the next generation. Notice that the $\Gamma^{\mathbf{p}\to\mathbf{c}}$ is computed only if the offspring solution is better than the parent solution.

The predefined value, γ, controls the maximum allowable $\Gamma^{\mathbf{p}\to\mathbf{c}}$. As we mentioned before, a high value of $\Gamma^{\mathbf{p}\to\mathbf{c}}$ may indicate the existence of similar solutions in the next generation. The lower the value we set for γ, the less tolerance the algorithm has for the diversity loss (the shrink of the solution spread). If we set the γ to be extremely low, we would expect that the diversity loss of the evolutionary search to be low also. However, this is likely to cause the disturbance of the algorithm's convergence to the true POF.

C. Genetic Operator

Implementation of the proposed online diversity metric on a selection operator aims to preserve the diversity of solutions throughout the search process. The proposed metric can also be used as an indicator of the population diversity during the search process. This indicator can provide useful information to the genetic operators (mutation and crossover operators) and the operators can adapt its parameters accordingly. We implement the metric as a diversity indicator and use the indicator to tune the parameters in the genetic operators.

When the diversity loss is high, we would like the operators to be more explorative to avoid the population trapping in a local POF. To implement such a scheme on the genetic operators, a precise definition of the high diversity loss is

required. Since the diversity loss of any algorithm during the search process is problem-dependent and algorithm-dependent, it would be improper if we preset a specific value to judge whether the diversity loss is high or not. To circumvent this problem, we propose to model the diversity loss trend to judge the current diversity loss level.

Since genetic operators react based on the diversity loss trend model and the calculated $\Gamma^{\mathbf{p}\to\mathbf{c}}$, the modeling method and the mechanism of the genetic operators have to be designed carefully. As any modeling method inevitably introduces certain level of uncertainty, the mechanism of the genetic operators should not be too sensitive to the modeling error. Otherwise, the modeling error may disrupt the search effort of the evolutionary algorithm. With these considerations, we would like the modeling approach to be conservative and genetic operators vary their explorative feature based on the certainty of the diversity loss.

1) Modeling the Trend of Diversity Loss: Least squares estimation is used to model the trend of the diversity loss due to its conceptual simplicity. From (7) and (10), we know that the proposed metric measures the solutions spread ratio between two solutions in consecutive generations. At the early evolutionary search process, the solutions generally scatter around the objective space. As the search progresses, the solutions approach the POF. This in turn reduces the spread and area occupied by the population. The reduction of the spread and area most likely results in high value of $\Gamma^{\mathbf{p}\to\mathbf{c}}$ in the early generation. At the late generation, the reduction of the spread is generally smaller than early generation due to smaller improvement of objective values. This results in smaller $\Gamma^{\mathbf{p}\to\mathbf{c}}$ than the former case. Exponential function can be used to model these characteristics, it is used in this paper. Mathematically, it can be expressed as

$$y = ae^{bx} \tag{12}$$

where x, y denote the generation number and moving average of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ at xth generation; a and b are the parameters to control the shape of the trend. $\mathbb{E}[\cdot]$ is an expectation (over individuals in population) operator. The moving average of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ is calculated based on past n_{mov} values of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$. Since the least squares method is sensitive to the outliers in the data, additional data filtering process is required. Moving average is used to filter out the noise and reduce the effect of outliers in the data. Taking the logarithm of both sides, we can obtain

$$\ln y = \ln a + bx. \tag{13}$$

We are ready to apply the least squares estimation method at this stage. The past moving average of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ and corresponding generation numbers are used to calculate the least squares estimation of parameters a and b

$$\begin{bmatrix} \ln y_1 \\ \ln y_2 \\ \vdots \\ \ln y_{i-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ \vdots & \vdots \\ 1 & i-1 \end{bmatrix} \begin{bmatrix} \ln a \\ b \end{bmatrix}$$
$$\mathbf{Y} = \mathbf{\Phi}\mathbf{\Lambda}$$
$$\mathbf{\Lambda} = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\mathbf{Y} \tag{14}$$

where i is the current generation number, y_i is the moving average of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ at generation i. For each generation, we use the estimated a and b to predict current $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$. If the current $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ is higher than the predicted value, the genetic operators adapt their parameters in hope of curbing the high diversity loss. To reduce the computation of the algorithm, an adaptive least squares method can be used [52].

2) Crossover Operator: Simulated binary crossover (SBX) operator is used as the properties of this operator are well investigated in [53]–[55]. The probability distribution of the offspring solution generated by SBX is defined as

$$c(\beta) = \begin{cases} 0.5(\eta_c+1)\beta^{\eta_c} & \text{if } \beta \le 1 \\ 0.5(\eta_c+1)\frac{1}{\beta^{\eta_c+2}} & \text{if } \beta > 1 \end{cases} \tag{15}$$

where β is the spread factor and η_c is the distribution index. Distribution index (η_c) plays an important role in controlling the spread of the offspring solutions. A large value of η_c causes higher probability of generating near parent solutions whereas a low value of η_c gives distant offspring solutions [55].

Based on these properties, we develop the following scheme. When the diversity loss is high, we reduce the value of distribution index (η_c) such that the offspring solution is different from the parent solutions. Otherwise, the algorithm increases the value of η_c. More precisely

$$\eta_{c,t+1} = \begin{cases} \eta_{c,t} - \Delta\eta_c & \text{if } \Gamma_{t+1}^{\mathbf{p}\to\mathbf{c}} > \hat{\Gamma}_{t+1}^{\mathbf{p}\to\mathbf{c}} \\ \eta_{c,t} + \Delta\eta_c & \text{otherwise} \end{cases} \tag{16}$$

where $\Delta\eta_c$ is the step size for the increment of η_c and $\hat{\Gamma}_{t+1}^{\mathbf{p}\to\mathbf{c}}$ is the prediction of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ at generation $t+1$. Notice that update rule in (16) may cause η_c to fall into the nonpositive region. Extra check therefore must be taken to ensure that η_c is greater than a certain positive value. In other words, there is a minimum bound for the η_c.

3) Mutation Operator: The polynomial mutation used in MOEA/D adds small perturbation to the offspring solutions and it is used to enhance exploitation ability of the algorithm. When the two parent solutions are close to each other, it is difficult to generate an offspring solution that is distant from the parents. To alleviate this problem, additional mutation operator is used. Gaussian noise is added to the offspring solution to enhance the diversity of the population especially when the diversity loss is high. If the diversity loss is lower than the estimated loss ($\hat{\Gamma}^{\mathbf{p}\to\mathbf{c}}$), the standard deviation of the Gaussian noise is set to zero (i.e., no Gaussian noise is added to the solution). This means

$$P_{n,t} = \begin{cases} \eta_p\left(\Gamma_t^{\mathbf{p}\to\mathbf{c}} - \hat{\Gamma}_t^{\mathbf{p}\to\mathbf{c}}\right) & \text{if } \Gamma_t^{\mathbf{p}\to\mathbf{c}} > \hat{\Gamma}_t^{\mathbf{p}\to\mathbf{c}} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

where $P_{n,t}$ is the standard deviation of the Gaussian noise at generation t and η_p is a predefined standard deviation increment rate. The increment rate affects the amplitude of noise injected into the population.

V. Experiment Results

We first examine the relationship between the proposed online diversity metric and the actual solution distribution in the objective space. The proposed online diversity metric

Algorithm 4 Decomposition-Based Multiobjective Evolutionary Algorithm With Online Diversity Indicator

Require:
MOP
A stopping criterion
N: Population size
T: The number of the weight vectors in the neighborhood of each weight vector
γ: Maximum allowable $\Gamma^{\mathbf{p}\to\mathbf{c}}$
Ensure:
Approximated POF $\{\mathbf{f}^1, \ldots, \mathbf{f}^N\}$
Approximated POS $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$
Step 1 ➢ Initialization:
Step 1.1 Generate evenly spread weight vectors. Find the T closest weight vectors (in terms of Euclidean distance) for each vector. Set $B(i) = i_1, \ldots, i_T$, where $\lambda^{i_1}, \ldots, \lambda^{i_T}$ are the T closest weight vectors to λ^i.
Step 1.2 Generate an initial population, $\mathbf{x}^1, \ldots, \mathbf{x}^N$, by uniformly random sampling the decision space. Evaluate each solution and set $\mathbf{f}^i = \mathbf{f}(\mathbf{x}^i)$. This forms the initial population, P_0.
Step 1.3 Initialize $\mathbf{z}$ by setting $z_k = \min_{j=1,\ldots,N} f_k^j$ where $k = 1, \ldots, m$.
Step 2 ➢ Update: Set $P_{\text{conv}} = C_{\text{conv}} = \emptyset$. For $i = 1, \ldots, N$, do
Step 2.1 Reproduction: Randomly select two indices from $B(i)$ and use SBX operator to produce a new solution. Apply polynomial mutation and proposed mutation to the new solution to produce $\mathbf{y}$.
Step 2.2 Update of z: Evaluate $\mathbf{y}$ to get $\mathbf{f}(\mathbf{y})$. If $f_j(\mathbf{y}) < z_j$ for any $j \in \{1, \ldots, m\}$, set $z_j = f_j(\mathbf{y})$.
Step 3 ➢ Environmental Selection: Pass the parent population (P_i) and offspring population (C_i) to Algorithm 2 to select the parent solutions for the next generation (P_{i+1}).
Step 4 ➢ Stopping Criterion: If the stopping criterion is satisfied, stop the process, output $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ and $\{\mathbf{f}^1, \ldots, \mathbf{f}^N\}$. Otherwise, go to **Step 2**.

is implemented in MOEA/D with population size $N = 100$ and generation $G = 500$. The algorithm is evaluated using WFG [56], CEC-09 [57] and TYD [47] test problems. In each generation, $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ is calculated by averaging $\Gamma^{\mathbf{p}\to\mathbf{c}}$ over the whole population. Then, moving average (with filter size, $n_{\text{mov}} = 10$) is performed to obtain smoother trend chart. Fig. 4 shows the estimated convergence directions using (3) and (7). Fig. 5 is one of the typical trend charts we obtained in our experiment. It can be noticed that the diversity loss is relatively high at the early generation and gradually decreases with the increase of generation number. The sub-figures show the actual solution distribution in the objective space at generations 2, 10, 20, 30, 50, 150, 200, and 400, respectively. If we compare the difference of solution distribution between generations 2 and 10 to the difference of solution distribution between generations 200 and 400, it can be observed that the former is more significant than the latter.

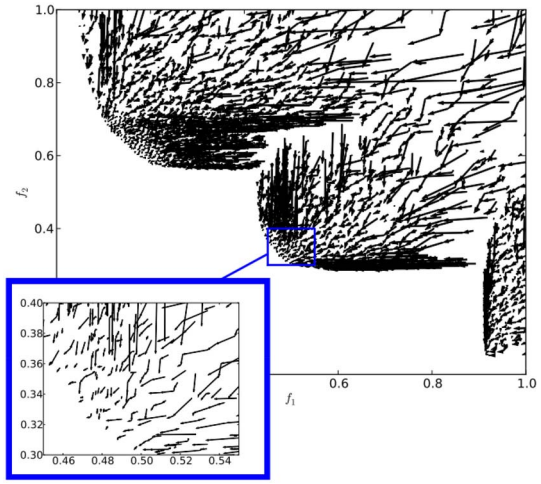

Fig. 4. Estimated convergence directions over 500 generations in CEC-09 UF1 benchmark test problem.

TABLE I
PARAMETER SETTINGS FOR EXPERIMENTS

Parameters	Values
Population size, N	100 or 300 (2 or 3 objectives)
Size of moving average, n_{mov}	10
Total number of generations	500
Total number of fitness evaluations	5×10^4 or 1.5×10^5
Neighbourhood size, T	20
Distribution index in SBX, η_c	20
Distribution index in mutation, η_m	20
Mutation rate, p_m	$1/n$ (n: decision variables)
Maximum allowable $\Gamma^{\mathbf{p}\to\mathbf{c}}$, γ	20
Step size of η_c, $\Delta\eta_c$	2
Std. dev. increment rate, η_p	0.5

In the following parts of this section, we present the optimization performance of the new selection and genetic operators that embed the concept of the proposed diversity metrics. All the test suites we used in the paper adopt the recommended settings as per [47], [56], and [57]. The proposed algorithm parameter settings are shown in Table I. inverted generational distance (IGD) [58], Hausdorff distance [22] and hypervolume [28] are used as the performance metrics to assess the optimization performance of the algorithm. The algorithm is also compared with other state-of-the-art algorithms in terms of optimization performance. The parameter settings of MOEAs used in the comparative study are shown in Table IV. At the end of this section, computational time of the algorithm is investigated and discussed.

A. Proposed Operators and Sensitivity of γ, $\Delta\eta_c$, and η_p

Tables II and III present the experiment results of the proposed operators using CEC-09 and WFG test suites, respectively. The first two rows of the tables show the optimization performance of MOEA/D (in terms of mean and standard deviation of IGD). IGD statistics of the proposed selection and genetic operators are also computed to compare

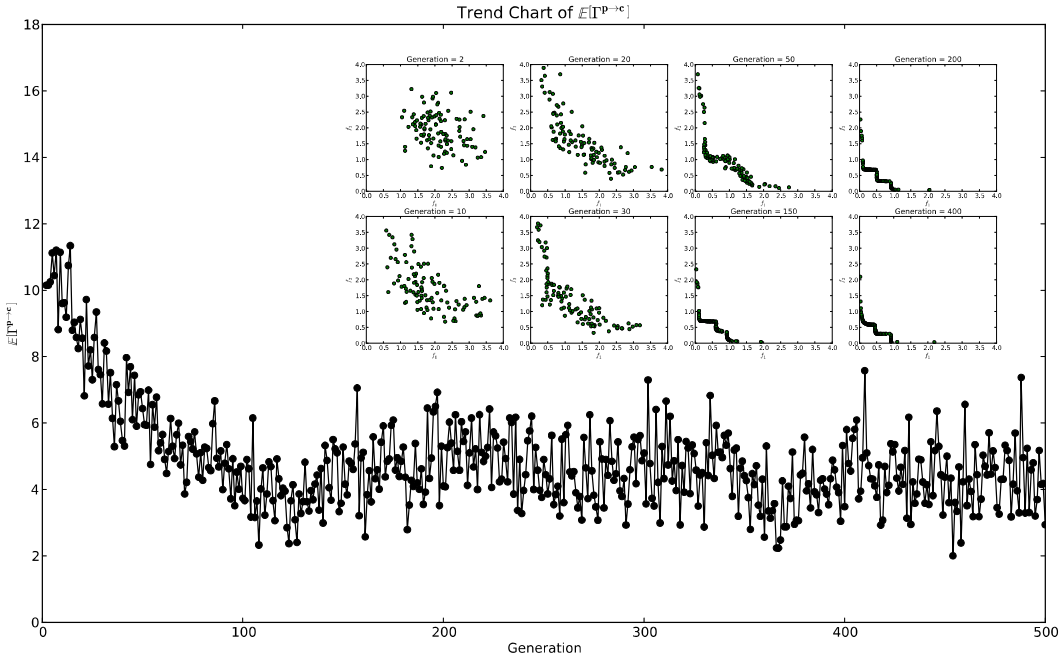

Fig. 5. Trend chart of 10-points-moving-average $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ over 500 generations under CEC-09 UF1 benchmark test problem. From the figure, it can be noticed that the diversity loss is high at the early generation and gradually decreases.

TABLE II
STATISTICS OF THE IGD VALUES (CEC-09 TEST SUITE)

	Algorithm	UF1	UF2	UF3	UF4	UF5	UF6	UF7	UF8	UF9	UF10
MOEA/D	Mean	0.1568	0.064	0.3064	0.05605	0.4318	0.3141	0.3536	0.148	0.1348	0.2937
(SBX)	Std. Dev.	0.06519	0.03101	0.02995	0.003423	0.08122	0.1212	0.1552	0.03579	0.06313	0.1303
New	Mean	**0.1016**	**0.03913**	**0.1968**	**0.05397**	0.4308	**0.2134**	**0.246**	**0.1213**	**0.1037**	**0.2322**
Selection	Std. Dev.	**0.04027**	**0.01359**	**0.04012**	**0.002614**	0.1057	**0.1476**	**0.1295**	**0.01895**	**0.04732**	**0.03176**
Operator	p (t-test)	**2.558E-4**	**2.492E-4**	**8.37E-17**	**0.0106**	0.9674	**0.005474**	**0.005121**	**7.823E-4**	**0.03549**	**0.01745**
	H_0 (t-test)	**Reject**	**Reject**	**Reject**	**Reject**	Accept	**Reject**	**Reject**	**Reject**	**Reject**	**Reject**
	p (r. sum)	**8.93E-5**	**3.88E-4**	**8.49E-10**	**0.012**	0.71	**1.64E-3**	**2.56E-3**	**1.84E-4**	0.104	5.01E-2
	H_0 (r. sum)	**Reject**	**Reject**	**Reject**	**Reject**	Accept	**Reject**	**Reject**	**Reject**	Accept	Accept
New	Mean	**0.0769**	0.03851	0.182	**0.04726**	**0.2534**	0.1604	**0.1264**	0.118	0.1076	0.2222
Genetic	Std. Dev.	**0.01417**	0.007949	0.04814	**0.001811**	**0.07322**	0.06848	**0.1205**	0.03232	0.04408	0.07258
Operator	p (t-test)	**3.114E-3**	0.8317	0.2001	**6.082E-16**	**6.728E-10**	0.1653	**4.75E-4**	0.6269	0.7286	0.4905
	H_0 (t-test)	**Reject**	Accept	Accept	**Reject**	**Reject**	Accept	**Reject**	Accept	Accept	Accept
	p (r. sum)	**1.53E-2**	0.322	**4.13E-2**	**8.56E-11**	**9.67E-9**	0.425	**2.92E-4**	**2.56E-2**	0.6152	8.11E-2
	H_0 (r. sum)	**Reject**	Accept	**Reject**	**Reject**	**Reject**	Accept	**Reject**	**Reject**	Accept	Accept

with MOEA/D. We perform statistical tests on the IGD values of each test problem by using paired two-sample student's t-test and Wilcoxon rank-sum test. The null hypothesis used in the test is that two sets of samples are insignificantly different as they are from similar probability distribution. The threshold chosen for statistical significance is 0.05.

From Table II, we notice that the proposed selection operator improves the IGD performance in most of CEC-09 test problems. The average IGD values of UF5, UF9, and UF10 test problem are insignificantly lower than MOEA/D as the null hypothesis is accepted by either t-test or rank-sum test. The average IGD values of the other CEC-09 test problems are significantly lower than MOEA/D (as the null hypothesis is rejected and the average IGD values are lower than MOEA/Ds). From Table III, no significant improvement of the average IGD value is observed even though the average IGD values are lower than MOEA/Ds. Comparing WFG and CEC-09 test suites, CEC-09 test problems require relatively more population diversity to avoid trapping in local POF. Without the proposed selection operator, MOEA/D can be easily trapped in a local POF because there is no mechanism to prevent existence of multiple identical individuals. For the WFG problems, MOEA/D is able to maintain the population diversity throughout the evolution (except for the WFG2). Hence, the effects of the proposed selection operator is not significant as compared to CEC-09 test problems.

Next, we evaluate the IGD performance of the proposed genetic operator. The proposed genetic operator adapts the

TABLE III
STATISTICS OF THE IGD VALUES (WFG TEST SUITE)

	Algorithm	WFG1	WFG2	WFG3	WFG4	WFG5	WFG6	WFG7	WFG8	WFG9
MOEA/D	Mean	1.048	0.1871	0.02033	0.01669	0.06907	0.082	0.02052	0.127	0.06063
(SBX)	Std. Dev.	0.04584	0.06433	0.005858	0.001525	0.0005596	0.02373	0.01106	0.009713	0.03807
New	Mean	1.043	**0.1553**	0.01846	0.0168	0.06919	0.0742	0.01714	0.1243	0.06528
Selection	Std. Dev.	0.05649	**0.0502**	0.00356	0.001091	0.0004146	0.023	0.006617	0.009893	0.04772
Operator	p (t-test)	0.6764	**0.03765**	0.1405	0.7381	0.3685	0.2011	0.1567	0.2918	0.678
	H_0 (t-test)	Accept	**Reject**	Accept	Accept	Accept	Accept	Accept	Accept	Accept
	p (r. sum)	0.7562	**1.10E-2**	0.1691	0.1087	**2.2E-2**	9.48E-2	0.322	0.535	0.579
	H_0 (r. sum)	Accept	**Reject**	Accept	Accept	**Reject**	Accept	Accept	Accept	Accept
New	Mean	**0.9452**	**0.06716**	0.01792	**0.01625**	**0.0687**	0.08491	0.01839	0.1235	0.05428
Genetic	Std. Dev.	**0.02332**	**0.0597**	0.00423	**0.001029**	**0.0002835**	0.03715	0.008918	0.006765	0.04172
Operator	p (t-test)	**1.095E-10**	**7.243E-8**	0.5943	**0.04628**	**2.138E-6**	0.1856	0.5376	0.7287	0.346
	H_0 (t-test)	**Reject**	**Reject**	Accept	**Reject**	**Reject**	Accept	Accept	Accept	Accept
	p (r. sum)	**6.264E-8**	**1.931E-5**	0.1171	**5.698E-3**	**5.186E-7**	0.1316	0.5154	0.7117	0.442
	H_0 (r. sum)	**Reject**	**Reject**	Accept	**Reject**	**Reject**	Accept	Accept	Accept	Accept

TABLE IV
PARAMETER SETTINGS OF MOEAS[1]

Parameter Settings	MOEA/D (DE)	MOEA/D (SBX)	NSDE	NSGA-II (SBX)	CCPSO	SPEA2	HyPE	PICEA-g
Differential weight, F	0.5	×	0.5	×	×	×	×	×
Crossover prob., p_c	1.0	1.0	1.0	1.0	1.0	0.8	1.0	1.0
Dist. index, η_c	×	20	×	20	×	×	15	15
Dist. index, η_m	20	20	20	20	×	×	20	20
Mutation rate, p_m	$1/n$	$1/n$	$1/n$	$1/n$	$1/n$	$1/(n \times n_{bit})$	1.0	1/n
Bit per decision var., n_{bit}	×	×	×	×	×	15	×	×
Neigh. prob. (MOEA/D), δ	0.9	×	0.9	×	×	×	×	×
Neigh. size(MOEA/D), T	20	×	20	×	×	×	×	×
Update limit (MOEA/D), n_r	2	×	2	×	×	×	×	×
Archive pop. size (2/3 objs)	100/300	100/300	100/300	100/300	100/300	100/300	100/300	100/300
Sub-pop. size (2/3 objs)	×	×	×	×	4(CEC-09, TYD), 5(WFG)/15	10/30	×	×
Total generation number	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)	500/ 300(TYD)
No. fitness eval. (2/3 objs)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)	5×10^4/ 1.5×10^5 (3×10^4 for TYD)
Inertia weight, niche radius, turbulence prob.	×	×	×	×	0.4, dynamic sharing, 0.1	×	×	×
No. goal vector (2/3 objs), goal vector bounds	×	×	×	×	×	×	×	100/300, 1.2×nadir pts (upper) and ideal vect. (lower)[1]
Hypervolume calculation	×	×	×	×	×	×	exact hyperv. calc.	×

operator's parameters by using $\mathbb{E}\left[\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}\right]$ and $\hat{\Gamma}^{\mathbf{p}\rightarrow\mathbf{c}}$. To obtain the $\mathbb{E}\left[\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}\right]$, the proposed selection operator is required to be implemented on the algorithm. For a fair comparison, we compare the IGD performance of the genetic operator with the selection operator. From Table II, we can observe that the genetic operator performs significantly better than the proposed selection operator in UF1, UF4, UF5, and UF7 test problems. For other test problems, the average IGD values are insignificantly different from the proposed selection operator's counterpart. From Table III, the genetic IGD performance is

[1]The approximated nadir point and ideal vector are used in the algorithm.

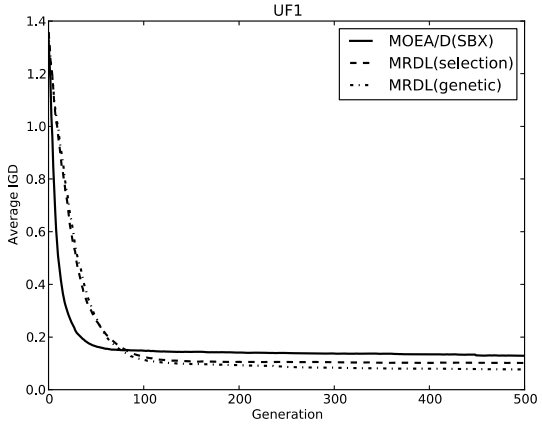

Fig. 6. Run-time performance of the MOEA/D (SBX), MOEA/D with proposed selection operator, and MOEA/D with proposed genetic operator.

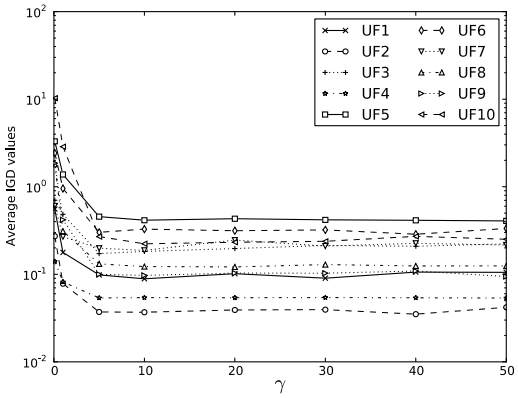

Fig. 7. Sensitivity of the algorithm's maximum allowable $\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}$ (γ) in CEC-09 test suite.

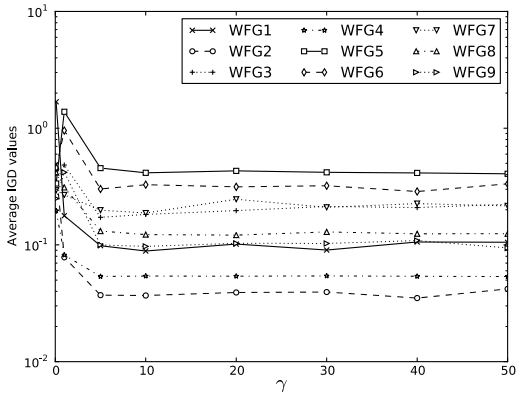

Fig. 8. Sensitivity of the algorithm's maximum allowable $\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}$ (γ) in WFG test suite.

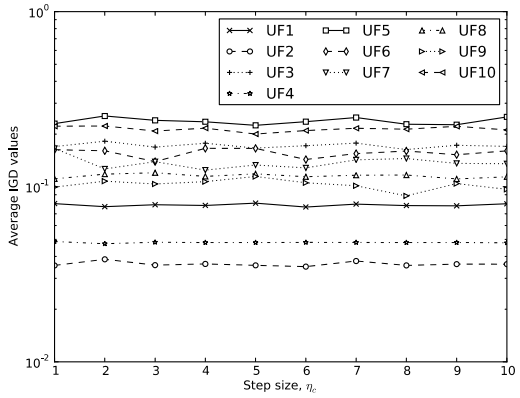

Fig. 9. Sensitivity of the algorithm's η_c in CEC-09 test suite.

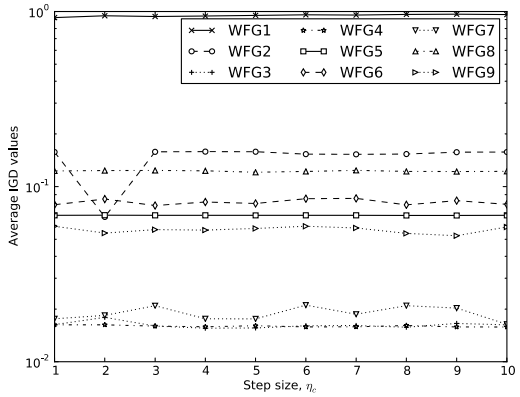

Fig. 10. Sensitivity of the algorithm's η_c in WFG test suite.

significantly better than the proposed selection operator in WFG1, WFG2, and WFG5 test problems (for WFG4, the p value is near to the threshold 0.05). In Tables II and III, all the statistically better results are shown in bold font. Fig. 6 shows the run-time performance of the proposed operators. The plot is generated based on IGD values of every generation (over 30 independent simulation runs). From the plot, we can observe that the convergence speed of the proposed algorithm is slow at the initial generations. This is probably because of the proposed selection operator, which controls the diversity loss of the population and prevents some parent-offspring replacements. If we compare the overall performance of the algorithms, we can observe that the proposed algorithm can achieve better performance in the long run. Similar plots have been observed in other benchmark problems. Based on the simulation results, we postulate that the proposed online diversity metric can be used to enhance the optimization performance of the MOEA/D.

Subsequently, the sensitivities of the maximum allowable $\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}$ (or its equivalence, γ), $\Delta\eta_c$, and η_p are investigated, respectively. Thirty independent simulation runs are performed and the average values of the IGD are recorded down as shown in Figs. 7–12. From Figs. 7 and 8, it can be observed that the $\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}$ threshold value, γ, should be set properly as extremely small value of γ may cause high value of IGD (poor optimization performance). On the other hand, the high value of γ results in the degeneration of the proposed selection operator into typical selection operator in MOEA/D. Based on numerous experiments, there is no significant difference between the proposed selection operator and MOEA/D when γ is higher than 1000. It is suggested that the value of γ should be set within the range of 5 to 50 for typical MOEA settings (population size, $n = 100 \sim 300$ and number of objectives, $m = 2 \sim 3$). From Figs. 9 and 10, it can be observed that the step size of η_c is rather insensitive in the range of $1 - 10$. Generally, low values of $\Delta\eta_c$ perform slightly better than high

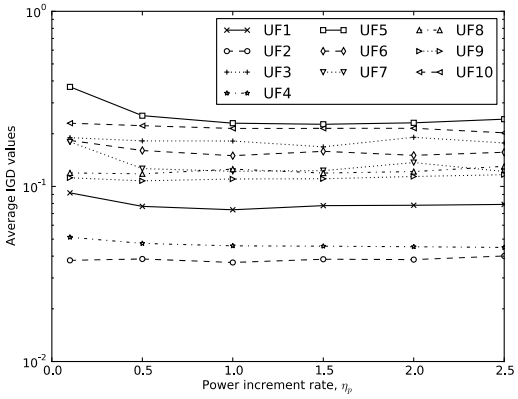

Fig. 11. Sensitivity of the algorithm's η_p in CEC-09 test suite.

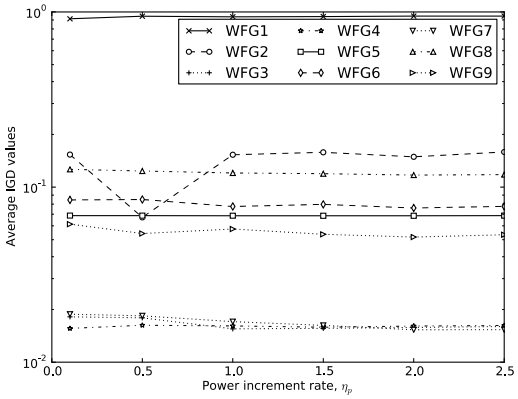

Fig. 12. Sensitivity of the algorithm's η_p in WFG test suite.

values of $\Delta\eta_c$. For the standard deviation increment rate (η_p), the high value is more desirable especially for CEC-09 test suite. This is because low values of η_p may not able to energize the offspring solution jump out of local POF. Based on our extensive experiments, we observed that the mutation operator is very helpful when the $\mathbb{E}\left[\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}\right]$ fluctuates around a particular value. This is often an indication that the whole population is either converging to global POF or trapping in local POF. Since the MOEA/D framework preserves good solutions that are found, generating distant offspring solution will not cause losing of good found solutions. If the whole population is in a local POF, generating distant solutions may be able to help the search process overcome the local trap.

B. Performance Comparison With Other MOEAs

From Tables II and III, it can be observed that the optimization performance of the MOEA/D is enhanced by implementing the proposed selection and genetic operators. To validate the superior optimization performance of the new algorithm, the proposed algorithm is compared with several state-of-the-art algorithms.

1) NSGA-II (SBX) [11] is one of the most popular MOEAs proposed by Deb *et al.* This algorithm uses nondominated sorting to obtain the Pareto rank of the solutions. Crowding distance is used to maintain the diversity of the algorithm.

TABLE V
FREQUENCIES OF THE RANKS ON CEC-09, WFG AND TYD TEST SUITES USING HYPERVOLUME

Algorithm	Rank								
	1	2	3	4	5	6	7	8	9
MOEA/D (MRDL)	14	2	3	2	2	1	1	0	0
MOEA/D (DE)	2	2	4	3	7	2	4	0	1
MOEA/D (SBX)	0	5	2	3	3	5	2	5	0
NSDE	2	6	1	1	4	1	3	7	0
NSGA-II (SBX)	2	7	3	6	1	3	2	0	1
CCPSO	1	3	4	1	2	1	1	5	7
SPEA2	2	1	0	7	2	8	4	1	0
HyPE	1	0	6	3	2	2	6	4	1
PICEA-g	1	1	0	0	1	2	2	3	15

TABLE VI
FREQUENCIES OF THE RANKS ON CEC-09, WFG AND TYD TEST SUITES USING IGD

Algorithm	Rank								
	1	2	3	4	5	6	7	8	9
MOEA/D (MRDL)	12	8	2	3	0	0	0	0	0
MOEA/D (DE)	4	5	2	2	3	2	6	1	0
MOEA/D (SBX)	0	6	6	2	1	4	3	3	0
NSDE	2	4	1	1	4	3	4	5	1
NSGA-II (SBX)	3	2	4	7	6	2	1	0	0
CCPSO	0	2	5	4	2	2	1	4	5
SPEA2	2	1	0	3	6	8	5	0	0
HyPE	1	2	1	3	3	1	2	10	2
PICEA-g	1	0	0	0	0	2	3	2	17

TABLE VII
FREQUENCIES OF THE RANKS ON CEC-09, WFG AND TYD TEST SUITES USING HAUSDORFF DISTANCE WITH $p = 2$

Algorithm	Rank								
	1	2	3	4	5	6	7	8	9
MOEA/D (MRDL)	9	7	7	1	1	0	0	0	0
MOEA/D (DE)	0	5	2	4	4	3	5	2	0
MOEA/D (SBX)	3	1	6	3	2	3	3	4	0
NSDE	4	2	0	0	6	2	7	3	1
NSGA-II (SBX)	4	4	3	4	6	2	1	1	0
CCPSO	1	4	3	3	1	3	1	3	6
SPEA2	2	2	1	5	2	9	2	1	1
HyPE	1	2	2	5	3	2	2	7	1
PICEA-g	1	0	0	0	0	1	3	4	16

2) NSDE [51] is a variant of NSGA-II. The main difference between these two algorithms is that differential evolution (DE) mutation operator is used to generate offspring solutions instead of SBX operator.
3) SPEA2 [59] is proposed by Ziztler. It uses clustering method to preserve the shape of approximated POF.

TABLE VIII
STATISTICS OF THE HYPERVOLUME VALUES

Algorithm	UF1	UF2	UF3	UF4	UF5	UF6
MOEA/D (MRDL)	8.713 ± 0.247 (5)	**3.758 ± 0.078(1)**	6.939 ± 0.244 (4)	1.881 ± 0.006 (2)	**58.84 ± 1.418(1)**	125.3 ± 2.098 (3)
MOEA/D (DE)	8.905 ± 0.269 (3)	3.698 ± 0.165 (4)	7.263 ± 0.644 (2)	1.778 ± 0.028 (9)	53.58 ± 1.979 (7)	124.0 ± 3.370 (4)
MOEA/D (SBX)	8.311 ± 0.414 (8)	3.624 ± 0.146 (7)	6.466 ± 0.144 (8)	1.850 ± 0.013 (5)	55.38 ± 1.424 (6)	122.7 ± 2.588 (6)
NSDE	9.023 ± 0.094 (2)	3.749 ± 0.056 (2)	**8.202 ± 0.039(1)**	1.815 ± 0.023 (8)	55.55 ± 1.460 (5)	125.7 ± 2.134 (2)
NSGA-II (SBX)	8.647 ± 0.193 (6)	3.706 ± 0.071 (3)	6.770 ± 0.275 (6)	1.875 ± 0.004 (3)	57.27 ± 1.903 (2)	**125.9 ± 2.425(1)**
CCPSO	**9.032 ± 0.119(1)**	3.616 ± 0.063 (8)	6.842 ± 0.291 (5)	1.861 ± 0.010 (4)	56.97 ± 1.361 (3)	123.6 ± 2.167 (5)
SPEA2	8.643 ± 0.181 (7)	3.698 ± 0.071 (4)	6.578 ± 0.294 (7)	1.835 ± 0.009 (6)	55.87 ± 2.055 (4)	122.0 ± 2.624 (7)
HyPE	8.745 ± 0.242 (4)	3.661 ± 0.075 (6)	7.173 ± 0.142 (3)	1.819 ± 0.008 (7)	52.24 ± 3.037 (8)	121.5 ± 3.258 (8)
PICEA-g	5.324 ± 0.235 (9)	2.455 ± 0.144 (9)	4.229 ± 0.154 (9)	**2.911 ± 0.084(1)**	23.41 ± 1.590 (9)	76.08 ± 3.872 (9)

Algorithm	UF7	UF8	UF9	UF10	WFG1	WFG2
MOEA/D (MRDL)	10.19 ± 0.958 (5)	1608 ± 5.642 (3)	2295 ± 39.68 (4)	**60587 ± 440.8(1)**	**6.141 ± 0.108(1)**	**5.629 ± 0.057(1)**
MOEA/D (DE)	10.40 ± 0.933 (3)	**1611 ± 0.811(1)**	**2348 ± 2.394(1)**	58718 ± 250.4 (5)	4.982 ± 0.146 (5)	5.504 ± 0.129 (5)
MOEA/D (SBX)	8.920 ± 0.847 (8)	1607 ± 5.992 (4)	2269 ± 49.15 (6)	60040 ± 876.9 (3)	4.789 ± 0.299 (8)	5.530 ± 0.120 (4)
NSDE	**10.85 ± 0.348(1)**	1608 ± 1.967 (2)	2344 ± 3.954 (2)	56870 ± 435.5 (8)	4.800 ± 0.100 (7)	5.547 ± 0.042 (3)
NSGA-II (SBX)	9.619 ± 0.965 (7)	1599 ± 0.505 (6)	2247 ± 37.37 (7)	60465 ± 610.2 (2)	4.649 ± 0.450 (9)	5.578 ± 0.036 (2)
CCPSO	10.75 ± 0.380 (2)	1512 ± 42.78 (8)	2207 ± 36.60 (8)	58584 ± 268.3 (6)	5.823 ± 0.190 (3)	4.609 ± 0.236 (9)
SPEA2	10.01 ± 0.882 (6)	1601 ± 4.134 (5)	2270 ± 65.35 (5)	59386 ± 653.8 (4)	5.487 ± 0.086 (4)	5.095 ± 0.108 (8)
HyPE	10.26 ± 0.887 (4)	1594 ± 0.627 (7)	2310 ± 43.83 (3)	57082 ± 211.8 (7)	4.953 ± 0.025 (6)	5.180 ± 0.025 (7)
PICEA-g	5.856 ± 0.322 (9)	1255 ± 33.16 (9)	1811 ± 55.75 (9)	41662 ± 896.5 (9)	6.096 ± 0.136 (2)	5.345 ± 0.231 (6)

Algorithm	WFG3	WFG4	WFG5	WFG6	WFG7	WFG8
MOEA/D (MRDL)	**5.765 ± 0.031(1)**	**3.400 ± 0.006(1)**	**3.843 ± 0.004(1)**	2.965 ± 0.113 (3)	**4.132 ± 0.038(1)**	**4.885 ± 0.021(1)**
MOEA/D (DE)	5.743 ± 0.012 (4)	3.086 ± 0.034 (6)	3.837 ± 0.002 (5)	2.815 ± 0.176 (5)	4.118 ± 0.007 (3)	4.829 ± 0.064 (3)
MOEA/D (SBX)	5.746 ± 0.031 (2)	3.395 ± 0.010 (2)	3.842 ± 0.005 (2)	2.984 ± 0.058 (2)	4.112 ± 0.042 (4)	4.859 ± 0.029 (2)
NSDE	5.664 ± 0.010 (5)	2.952 ± 0.028 (8)	3.778 ± 0.016 (7)	2.807 ± 0.177 (6)	4.039 ± 0.009 (5)	4.657 ± 0.064 (5)
NSGA-II (SBX)	5.746 ± 0.010 (2)	3.378 ± 0.008 (3)	3.842 ± 0.005 (2)	**3.018 ± 0.035(1)**	4.127 ± 0.008 (2)	4.780 ± 0.018 (4)
CCPSO	5.003 ± 0.175 (9)	2.852 ± 0.198 (9)	3.417 ± 0.160 (9)	2.727 ± 0.095 (8)	2.999 ± 0.360 (9)	3.721 ± 0.262 (9)
SPEA2	5.311 ± 0.150 (7)	3.240 ± 0.043 (4)	3.807 ± 0.015 (6)	2.903 ± 0.080 (4)	3.902 ± 0.100 (6)	4.635 ± 0.040 (6)
HyPE	5.292 ± 0.013 (8)	2.966 ± 0.009 (7)	3.839 ± 0.006 (4)	2.347 ± 0.181 (9)	3.736 ± 0.012 (8)	4.411 ± 0.023 (7)
PICEA-g	5.495 ± 0.082 (6)	3.199 ± 0.051 (5)	3.602 ± 0.112 (8)	2.794 ± 0.109 (7)	3.846 ± 0.163 (7)	4.375 ± 0.103 (8)

Algorithm	WFG9	TYD1	TYD2	TYD3	TYD4	TYD5
MOEA/D (MRDL)	**6.135 ± 0.231(1)**	**8.554 ± 0.012(1)**	**1184 ± 0.021(1)**	4747 ± 1.674 (2)	**5882 ± 1.743(1)**	18172 ± 21.31 (6)
MOEA/D (DE)	6.128 ± 0.164 (2)	8.097 ± 0.574 (5)	1180 ± 11.65 (5)	4679 ± 31.59 (7)	5726 ± 131.2 (7)	18154 ± 12.59 (7)
MOEA/D (SBX)	6.093 ± 0.209 (3)	7.649 ± 0.549 (7)	1167 ± 15.53 (8)	4694 ± 29.77 (6)	5796 ± 64.32 (6)	18175 ± 19.94 (5)
NSDE	5.802 ± 0.206 (7)	8.475 ± 0.292 (2)	1181 ± 1.598 (4)	4621 ± 20.35 (8)	5622 ± 82.59 (8)	17965 ± 105.4 (8)
NSGA-II (SBX)	5.969 ± 0.294 (4)	8.165 ± 0.565 (4)	1182 ± 8.969 (2)	4733 ± 23.43 (4)	5834 ± 61.93 (5)	18188 ± 18.71 (4)
CCPSO	5.456 ± 0.330 (9)	7.260 ± 0.747 (8)	1177 ± 13.46 (7)	4743 ± 7.360 (3)	5875 ± 24.75 (2)	18196 ± 6.441 (2)
SPEA2	5.839 ± 0.209 (6)	8.071 ± 0.560 (6)	1179 ± 8.930 (6)	**4749 ± 1.331(1)**	5847 ± 33.91 (4)	**18197 ± 16.15(1)**
HyPE	5.863 ± 0.183 (5)	8.297 ± 0.490 (3)	1181 ± 13.58 (3)	4728 ± 38.45 (5)	5866 ± 70.37 (3)	18196 ± 14.03 (3)
PICEA-g	5.599 ± 0.133 (8)	6.003 ± 0.345 (9)	268.1 ± 90.71 (9)	551.9 ± 251.1 (9)	865.0 ± 275.0 (9)	2185 ± 872.2 (9)

4) CCPSO [60] is proposed by Goh *et al.* and it utilizes the cooperative co-evolutionary mechanism and particle swarm optimization (PSO) algorithm to solve the MOP.
5) MOEA/D (DE) [51] is a variant of MOEA/D and it uses DE mutation operator instead of SBX crossover operator to produce offspring solutions.
6) HyPE [61] is a hypervolume-based MOEA that utilizes hypervolume indicator to improve the search process.[2]
7) PICEA-g [62] is a MOEA that co-evolves goal vectors with candidate solutions.[3]

Tables VIII–X show the hypervolume, IGD, and Hausdorff distance statistics of different MOEAs. The values of each cell denote the hypervolume (IGD or Hausdorff distance) mean and standard deviation for a particular MOEA evaluated on a particular test problem. The integer value inside the bracket refers to the rank of the MOEA with respect to a particular test problem. The values shown in boldface represent the best performance result of a particular problem among different MOEAs. Tables V–VII present the rank frequency of different MOEAs. From these tables, it can be observed that the proposed algorithm ranks well in most of the benchmark problems. The performance of PICEA-g in CEC-09 and TYD test suites is not as expected. Based on our observation, the algorithm stagnates on certain benchmark problems (TYD2-TYD5, UF1-3, UF5-6) that demand high diversity of population. However, this does not imply that the algorithm is inferior to other MOEAs. In fact, PICEA-g and HyPE perform well on many-objective optimization problems

[2]The HyPE code can be downloaded from the ETH Zurich website (http://www.tik.ee.ethz.ch/sop/download/supplementary/hype/). Default parameter configuration is used in the comparative study.

[3]The PICEA-g code can be downloaded from the University of Sheffield website (http://www.sheffield.ac.uk/acse/staff/rstu/ruiwang/index). Default parameter configuration is used in the comparative study.

TABLE IX
STATISTICS OF THE IGD VALUES

	UF1	UF2	UF3	UF4	UF5	UF6
MOEA/D (MRDL)	0.077 ± 0.014 (4)	**0.039 ± 0.008(1)**	0.182 ± 0.048 (3)	**0.047 ± 0.002(1)**	**0.253 ± 0.073(1)**	0.160 ± 0.068 (2)
MOEA/D (DE)	**0.047 ± 0.037(1)**	0.043 ± 0.032 (2)	**0.151 ± 0.069(1)**	0.087 ± 0.010 (8)	0.764 ± 0.131 (6)	0.340 ± 0.115 (4)
MOEA/D (SBX)	0.157 ± 0.065 (8)	0.064 ± 0.031 (7)	0.306 ± 0.030 (7)	0.056 ± 0.003 (3)	0.432 ± 0.081 (3)	0.314 ± 0.121 (3)
NSDE	0.060 ± 0.016 (3)	0.043 ± 0.005 (2)	0.152 ± 0.027 (2)	0.072 ± 0.008 (6)	0.849 ± 0.170 (7)	0.396 ± 0.087 (6)
NSGA-II (SBX)	0.123 ± 0.032 (6)	0.048 ± 0.012 (4)	0.218 ± 0.067 (4)	0.053 ± 0.002 (2)	0.326 ± 0.094 (2)	**0.149 ± 0.077(1)**
CCPSO	0.048 ± 0.011 (2)	0.048 ± 0.008 (4)	0.302 ± 0.039 (5)	0.064 ± 0.007 (4)	0.454 ± 0.073 (4)	0.430 ± 0.025 (7)
SPEA2	0.134 ± 0.041 (7)	0.063 ± 0.007 (6)	0.302 ± 0.041 (5)	0.068 ± 0.003 (5)	0.474 ± 0.088 (5)	0.358 ± 0.109 (5)
HyPE	0.105 ± 0.025 (5)	0.071 ± 0.004 (8)	0.412 ± 0.013 (8)	0.085 ± 0.003 (7)	1.118 ± 0.203 (8)	0.531 ± 0.099 (8)
PICEA-g	1.000 ± 0.096 (9)	0.472 ± 0.045 (9)	0.949 ± 0.066 (9)	0.223 ± 0.030 (9)	4.551 ± 0.329 (9)	4.458 ± 0.424 (9)
	UF7	UF8	UF9	UF10	WFG1	WFG2
MOEA/D (MRDL)	0.126 ± 0.120 (4)	0.118 ± 0.032 (2)	0.108 ± 0.044 (2)	**0.222 ± 0.073(1)**	0.945 ± 0.023 (2)	0.067 ± 0.060 (2)
MOEA/D (DE)	0.102 ± 0.165 (3)	**0.091 ± 0.012(1)**	**0.106 ± 0.046(1)**	0.583 ± 0.072 (6)	1.163 ± 0.014 (7)	0.167 ± 0.088 (4)
MOEA/D (SBX)	0.354 ± 0.155 (8)	0.148 ± 0.036 (3)	0.135 ± 0.063 (3)	0.294 ± 0.130 (2)	1.048 ± 0.046 (4)	0.187 ± 0.064 (5)
NSDE	**0.039 ± 0.042(1)**	0.152 ± 0.030 (4)	0.194 ± 0.065 (5)	2.431 ± 0.185 (7)	1.218 ± 0.005 (9)	**0.046 ± 0.025(1)**
NSGA-II (SBX)	0.236 ± 0.145 (7)	0.219 ± 0.010 (6)	0.165 ± 0.050 (4)	0.324 ± 0.070 (3)	1.079 ± 0.081 (5)	0.160 ± 0.028 (3)
CCPSO	0.096 ± 0.038 (2)	0.257 ± 0.053 (8)	0.289 ± 0.051 (6)	0.486 ± 0.030 (4)	0.976 ± 0.036 (3)	0.545 ± 0.146 (9)
SPEA2	0.169 ± 0.128 (6)	0.193 ± 0.036 (5)	0.294 ± 0.059 (7)	0.562 ± 0.107 (5)	1.086 ± 0.018 (6)	0.238 ± 0.028 (7)
HyPE	0.139 ± 0.117 (5)	0.243 ± 0.006 (7)	0.333 ± 0.052 (8)	2.708 ± 0.124 (8)	1.200 ± 0.003 (8)	0.202 ± 0.054 (6)
PICEA-g	1.060 ± 0.091 (9)	2.181 ± 0.195 (9)	2.218 ± 0.208 (9)	11.571 ± 0.934 (9)	**0.931 ± 0.027(1)**	0.277 ± 0.128 (8)
	WFG3	WFG4	WFG5	WFG6	WFG7	WFG8
MOEA/D (MRDL)	**0.018 ± 0.004(1)**	**0.016 ± 0.001(1)**	0.069 ± 0.000 (2)	0.085 ± 0.037 (3)	0.018 ± 0.009 (2)	**0.124 ± 0.007(1)**
MOEA/D (DE)	0.020 ± 0.002 (2)	0.081 ± 0.008 (7)	0.069 ± 0.000 (2)	0.107 ± 0.032 (5)	0.019 ± 0.001 (3)	0.127 ± 0.013 (2)
MOEA/D (SBX)	0.020 ± 0.006 (2)	0.017 ± 0.002 (2)	0.069 ± 0.001 (2)	0.082 ± 0.024 (2)	0.021 ± 0.011 (4)	0.127 ± 0.010 (2)
NSDE	0.034 ± 0.002 (5)	0.093 ± 0.004 (8)	0.075 ± 0.002 (7)	0.108 ± 0.035 (6)	0.031 ± 0.002 (5)	0.141 ± 0.010 (5)
NSGA-II (SBX)	0.021 ± 0.002 (4)	0.019 ± 0.001 (3)	0.071 ± 0.000 (5)	**0.064 ± 0.007(1)**	**0.017 ± 0.001(1)**	0.137 ± 0.006 (4)
CCPSO	0.183 ± 0.052 (9)	0.061 ± 0.033 (5)	0.090 ± 0.013 (8)	0.123 ± 0.049 (8)	0.212 ± 0.073 (9)	0.220 ± 0.049 (9)
SPEA2	0.104 ± 0.030 (7)	0.036 ± 0.005 (4)	0.073 ± 0.001 (6)	0.087 ± 0.016 (4)	0.051 ± 0.018 (6)	0.170 ± 0.011 (6)
HyPE	0.107 ± 0.003 (8)	0.097 ± 0.002 (9)	**0.068 ± 0.000(1)**	0.200 ± 0.036 (9)	0.086 ± 0.003 (8)	0.218 ± 0.006 (8)
PICEA-g	0.074 ± 0.022 (6)	0.065 ± 0.014 (6)	0.107 ± 0.016 (9)	0.112 ± 0.039 (7)	0.070 ± 0.022 (7)	0.210 ± 0.030 (7)
	WFG9	TYD1	TYD2	TYD3	TYD4	TYD5
MOEA/D (MRDL)	**0.054 ± 0.042(1)**	**0.012 ± 0.001(1)**	**0.047 ± 0.001(1)**	0.348 ± 0.104 (2)	**0.035 ± 0.010(1)**	**0.140 ± 0.103(1)**
MOEA/D (DE)	0.060 ± 0.029 (2)	0.069 ± 0.071 (5)	0.195 ± 0.553 (5)	2.089 ± 0.636 (7)	1.779 ± 1.505 (7)	1.603 ± 0.562 (7)
MOEA/D (SBX)	0.061 ± 0.038 (3)	0.133 ± 0.091 (7)	0.784 ± 0.760 (8)	1.906 ± 0.567 (6)	0.741 ± 0.533 (6)	1.105 ± 0.629 (6)
NSDE	0.111 ± 0.038 (7)	0.027 ± 0.038 (2)	0.075 ± 0.028 (2)	2.090 ± 0.326 (8)	2.431 ± 0.892 (8)	1.606 ± 0.282 (8)
NSGA-II (SBX)	0.084 ± 0.053 (4)	0.057 ± 0.074 (4)	0.141 ± 0.437 (3)	0.720 ± 0.670 (5)	0.380 ± 0.450 (5)	0.792 ± 0.555 (5)
CCPSO	0.152 ± 0.053 (9)	0.253 ± 0.158 (8)	0.287 ± 0.233 (6)	0.491 ± 0.263 (3)	0.289 ± 0.166 (3)	0.250 ± 0.310 (3)
SPEA2	0.107 ± 0.039 (6)	0.071 ± 0.072 (6)	0.293 ± 0.392 (7)	**0.199 ± 0.101(1)**	0.290 ± 0.262 (4)	0.209 ± 0.575 (2)
HyPE	0.102 ± 0.035 (5)	0.050 ± 0.061 (3)	0.145 ± 0.597 (4)	0.626 ± 0.796 (4)	0.235 ± 0.815 (2)	0.401 ± 0.523 (4)
PICEA-g	0.147 ± 0.019 (8)	0.571 ± 0.049 (9)	23.825 ± 4.336 (9)	62.566 ± 8.198 (9)	65.691 ± 7.000 (9)	124.355 ± 12.339 (9)

as they offer comparability between alternative solutions in high dimensional objective space [61], [62].

C. Computational Time

The introduction of the proposed online diversity method incurs additional computational time due to calculation of $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$ for each individual in the population. The computation time of $\Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$ varies with the number of convergence directions stored in two convergence direction estimation set. In the worst case scenario, this involves $(1+2+\cdots+(N-1)) \approx N^2/2$ times calculation of triangular area for every generation.

An experiment is conducted to visualize the relationship between the average case computational time and the population size of the algorithm. In this experiment, only the population size of the MOEA is varied whereas other parameters are kept at their nominal values as shown in Table I. For each population size, ten independent runs are conducted and the average computational time is recorded. The final simulation results are shown in Fig. 13.

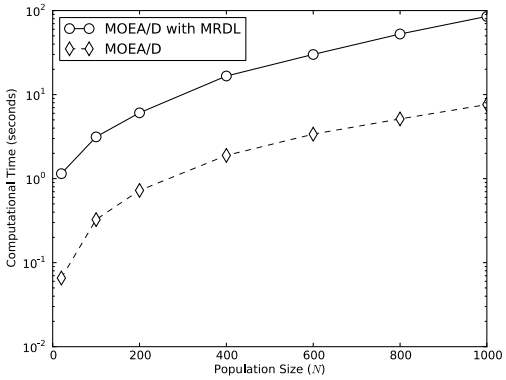

Fig. 13. Computational time for MOEA/D and MOEA/D with proposed selection operator. Each point in the graph denotes the average computation time over 10 independent runs.

TABLE X
STATISTICS OF THE HAUSDORFF DISTANCE WITH $p = 2$

	UF1	UF2	UF3	UF4	UF5	UF6
MOEA/D (MRDL)	0.098 ± 0.028 (3)	**0.056 ± 0.018(1)**	0.214 ± 0.051 (3)	**0.048 ± 0.002(1)**	**0.327 ± 0.104(1)**	0.229 ± 0.075 (2)
MOEA/D (DE)	0.081 ± 0.061 (2)	0.069 ± 0.053 (4)	0.193 ± 0.085 (2)	0.091 ± 0.010 (7)	0.960 ± 0.279 (6)	0.427 ± 0.136 (5)
MOEA/D (SBX)	0.198 ± 0.081 (8)	0.106 ± 0.054 (8)	0.365 ± 0.035 (7)	0.058 ± 0.004 (3)	0.488 ± 0.098 (3)	0.386 ± 0.149 (3)
NSDE	0.148 ± 0.151 (5)	0.060 ± 0.016 (2)	**0.164 ± 0.029(1)**	0.080 ± 0.008 (6)	1.324 ± 0.449 (7)	0.492 ± 0.187 (7)
NSGA-II (SBX)	0.152 ± 0.040 (6)	0.068 ± 0.022 (3)	0.264 ± 0.078 (4)	0.056 ± 0.002 (2)	0.354 ± 0.101 (2)	**0.184 ± 0.096(1)**
CCPSO	**0.055 ± 0.015(1)**	0.069 ± 0.016 (4)	0.326 ± 0.041 (6)	0.068 ± 0.008 (4)	0.513 ± 0.086 (5)	0.484 ± 0.037 (6)
SPEA2	0.160 ± 0.048 (7)	0.078 ± 0.013 (6)	0.320 ± 0.043 (5)	0.073 ± 0.004 (5)	0.497 ± 0.101 (4)	0.405 ± 0.163 (4)
HyPE	0.124 ± 0.032 (4)	0.078 ± 0.008 (6)	0.436 ± 0.036 (8)	0.093 ± 0.004 (8)	1.450 ± 0.321 (8)	0.598 ± 0.199 (8)
PICEA-g	1.136 ± 0.094 (9)	0.537 ± 0.055 (9)	1.038 ± 0.086 (9)	1.684 ± 0.316 (9)	5.096 ± 0.277 (9)	5.552 ± 0.489 (9)
	UF7	UF8	UF9	UF10	WFG1	WFG2
MOEA/D (MRDL)	0.177 ± 0.162 (4)	0.314 ± 0.115 (3)	0.342 ± 0.126 (3)	1.208 ± 0.659 (3)	0.946 ± 0.023 (2)	0.123 ± 0.131 (2)
MOEA/D (DE)	0.142 ± 0.201 (3)	1.875 ± 1.079 (6)	2.581 ± 1.045 (8)	1.562 ± 0.361 (4)	1.165 ± 0.014 (7)	0.321 ± 0.154 (4)
MOEA/D (SBX)	0.456 ± 0.181 (8)	**0.216 ± 0.063(1)**	**0.190 ± 0.089(1)**	**0.484 ± 0.101(1)**	1.096 ± 0.058 (5)	0.372 ± 0.091 (7)
NSDE	**0.080 ± 0.071(1)**	0.777 ± 0.539 (5)	1.373 ± 1.306 (6)	4.906 ± 1.797 (7)	1.219 ± 0.005 (9)	**0.055 ± 0.057(1)**
NSGA-II (SBX)	0.316 ± 0.183 (7)	2.938 ± 0.270 (8)	0.294 ± 0.315 (2)	2.817 ± 4.482 (5)	1.150 ± 0.108 (6)	0.342 ± 0.061 (5)
CCPSO	0.121 ± 0.053 (2)	0.310 ± 0.054 (2)	0.344 ± 0.048 (4)	0.532 ± 0.048 (2)	0.977 ± 0.036 (3)	0.806 ± 0.208 (9)
SPEA2	0.227 ± 0.167 (6)	2.987 ± 0.959 (9)	2.530 ± 0.875 (7)	7.831 ± 5.306 (8)	1.087 ± 0.019 (4)	0.361 ± 0.045 (6)
HyPE	0.183 ± 0.154 (5)	0.326 ± 0.009 (4)	0.879 ± 0.646 (5)	3.844 ± 0.102 (6)	1.201 ± 0.003 (8)	0.310 ± 0.108 (3)
PICEA-g	1.199 ± 0.089 (9)	2.825 ± 0.225 (7)	2.847 ± 0.217 (9)	14.391 ± 1.329 (9)	**0.932 ± 0.027(1)**	0.464 ± 0.187 (8)
	WFG3	WFG4	WFG5	WFG6	WFG7	WFG8
MOEA/D (MRDL)	**0.021 ± 0.005(1)**	**0.019 ± 0.001(1)**	0.069 ± 0.000 (2)	0.098 ± 0.055 (3)	0.025 ± 0.021 (3)	0.158 ± 0.003 (2)
MOEA/D (DE)	0.023 ± 0.002 (2)	0.085 ± 0.009 (5)	0.070 ± 0.000 (3)	0.109 ± 0.030 (5)	0.021 ± 0.001 (2)	0.166 ± 0.011 (4)
MOEA/D (SBX)	0.025 ± 0.015 (4)	0.020 ± 0.004 (2)	0.070 ± 0.001 (3)	0.099 ± 0.044 (4)	0.029 ± 0.026 (4)	0.159 ± 0.007 (3)
NSDE	0.035 ± 0.002 (5)	0.095 ± 0.004 (7)	0.076 ± 0.002 (7)	0.109 ± 0.034 (5)	0.032 ± 0.002 (5)	0.176 ± 0.011 (5)
NSGA-II (SBX)	0.023 ± 0.002 (2)	0.022 ± 0.002 (3)	0.071 ± 0.001 (5)	**0.064 ± 0.007(1)**	**0.020 ± 0.002(1)**	**0.153 ± 0.005(1)**
CCPSO	0.276 ± 0.073 (9)	0.176 ± 0.097 (9)	0.112 ± 0.032 (8)	0.169 ± 0.072 (8)	0.275 ± 0.071 (9)	0.304 ± 0.075 (9)
SPEA2	0.104 ± 0.030 (6)	0.037 ± 0.005 (4)	0.074 ± 0.001 (6)	0.088 ± 0.016 (2)	0.052 ± 0.018 (6)	0.183 ± 0.008 (6)
HyPE	0.111 ± 0.003 (8)	0.098 ± 0.002 (8)	**0.068 ± 0.000(1)**	0.213 ± 0.038 (9)	0.086 ± 0.003 (7)	0.230 ± 0.005 (7)
PICEA-g	0.109 ± 0.047 (7)	0.088 ± 0.021 (6)	0.122 ± 0.027 (9)	0.132 ± 0.059 (7)	0.104 ± 0.040 (8)	0.235 ± 0.047 (8)
	WFG9	TYD1	TYD2	TYD3	TYD4	TYD5
MOEA/D (MRDL)	**0.060 ± 0.040(1)**	0.162 ± 0.018 (5)	**0.105 ± 0.003(1)**	0.467 ± 0.176 (2)	**0.050 ± 0.016(1)**	**0.239 ± 0.160(1)**
MOEA/D (DE)	0.067 ± 0.028 (2)	0.238 ± 0.079 (6)	0.322 ± 0.778 (5)	3.465 ± 1.075 (7)	2.037 ± 1.677 (7)	1.981 ± 0.608 (8)
MOEA/D (SBX)	0.075 ± 0.037 (3)	0.276 ± 0.148 (7)	1.321 ± 1.118 (8)	3.234 ± 0.956 (6)	1.093 ± 0.652 (5)	1.530 ± 0.629 (6)
NSDE	0.111 ± 0.038 (7)	**0.077 ± 0.082(1)**	0.113 ± 0.019 (2)	3.679 ± 0.595 (8)	2.678 ± 1.032 (8)	1.770 ± 0.376 (7)
NSGA-II (SBX)	0.086 ± 0.052 (4)	0.124 ± 0.165 (3)	0.276 ± 0.661 (4)	1.432 ± 1.351 (5)	0.592 ± 0.629 (4)	1.145 ± 0.644 (5)
CCPSO	0.182 ± 0.083 (9)	0.350 ± 0.199 (8)	0.703 ± 0.732 (7)	0.639 ± 0.472 (3)	1.100 ± 2.157 (6)	0.414 ± 0.523 (3)
SPEA2	0.107 ± 0.038 (6)	0.141 ± 0.158 (4)	0.470 ± 0.546 (6)	**0.298 ± 0.147(1)**	0.385 ± 0.348 (3)	0.268 ± 0.689 (2)
HyPE	0.103 ± 0.034 (5)	0.089 ± 0.117 (2)	0.235 ± 0.762 (3)	1.207 ± 1.504 (4)	0.276 ± 0.910 (2)	0.591 ± 0.700 (4)
PICEA-g	0.153 ± 0.020 (8)	0.629 ± 0.060 (9)	27.34 ± 4.465 (9)	66.48 ± 7.498 (9)	73.14 ± 7.898 (9)	139.8 ± 15.51 (9)

From Fig. 13, it is clear that the proposed method is computationally more expensive than a typical MOEA/D. The computational time difference becomes more significant when the population size increases. Although the proposed online diversity metric incurs additional computational cost, proper use of the metric in the algorithm design can enhance the optimization performance of the algorithm.

D. Correlation With IGD

Generally, the trend of $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ is decreasing with the increase of generation number. To understand how well the performance metric reflects the spread and distribution of the population along the POF, the correlation between IGD trend and n_{mov}-generation moving average $\mathbb{E}\left[\Gamma^{\mathbf{p}\to\mathbf{c}}\right]$ trend is computed. Figs. 14 and 15 show the correlation coefficient between these two metrics over different test problems. Each box represents the distribution of correlation coefficients obtained in 30 independent simulation runs. From the two figures, it is clear that the correlation between these two metrics is stronger in WFG test suite than CEC-09 test suite. The correlation is weaker in CEC-09 problems probably due to deceptive local POF where the newly generated offspring solution dominates the most of the population. This implies that the nondominated set only consists of a few solutions. Generally, this lasts for a number of generations and it implies that the IGD value is relatively constant. However, the $\Gamma^{\mathbf{p}\to\mathbf{c}}$ continues varying due to the generation of offspring solutions that dominate some other parent solutions. This situation causes the correlation between these two metrics becomes weaker. Based on our observation, this scenario rarely happened in WFG test problem. Although the correlation is weaker in

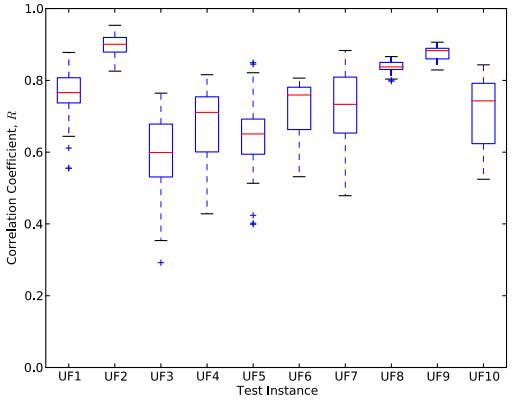

Fig. 14. Boxplot of the correlation coefficient between IGD trend and 10-points-moving-average $\mathbb{E}\left[\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}\right]$ trend in CEC-09 test problems. For each test problem, 30 simulation runs have been performed.

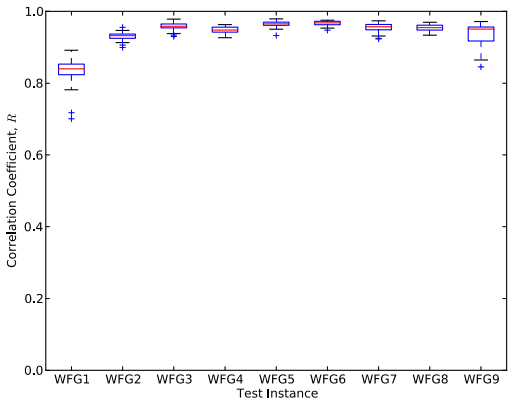

Fig. 15. Boxplot of the correlation coefficient between IGD trend and 10-points-moving-average $\mathbb{E}\left[\Gamma^{\mathbf{p}\rightarrow\mathbf{c}}\right]$ trend in WFG test problems. For each test problem, 30 simulation runs have been performed.

CEC-09 problems, majority of the correlation coefficient is higher than 0.6. This implies that the correlation of the online diversity metric with the IGD metric is moderate to high and the exact extent of correlation depends on the nature of the problems.

VI. Conclusion

This paper has proposed an online diversity metric with its implementation on decomposition-based MOEA. The diversity metric is defined based on the geometrical interpretation of convergence and diversity. The metric has been implemented on MOEA/D selection and genetic operators to demonstrate its applicability and usability. The simulation results have shown an improvement of the algorithm's optimization performance. We have also analyzed the additional computational time incurred by the proposed diversity metric. In real world optimization problems, fitness evaluation is often far more computationally expensive than the algorithm's operation. Good optimization performance of an algorithm often outweighs the algorithm's computation time. In this scenario, the use of online diversity can be justified.

References

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: Wiley, 2001.
[2] C. Coello, G. Lamont, and D. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, vol. 5. Boston, MA, USA: Springer, 2007.
[3] K. Miettinen, *Nonlinear Multiobjective Optimization*, vol. 12. Boston, MA, USA: Springer, 1999.
[4] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.
[5] K. Deb, "Genetic algorithms in multimodal function optimization," Master's thesis, Dept. Eng. Mech., Univ. Alabama, TCGA Rep. 89002, 1989.
[6] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," Master's thesis, Dept. Aeronaut. Astronaut., Massachusetts Inst. Technol., Cambridge, MA, USA, 1995.
[7] T. Okabe, Y. Jin, and B. Sendhoff, "A critical survey of performance indices for multi-objective optimisation," in *Proc. Congr. Evol. Comput.*, vol. 2. 2003, pp. 878–885.
[8] E. Zitzler, *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, vol. 63. Aachen, Germany: Shaker, 1999.
[9] K. C. Tan, T. H. Lee, and E. F. Khor, "Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons," *Artif. Intell. Rev.*, vol. 17, no. 4, pp. 251–290, 2002.
[10] J. Knowles and D. Corne, "On metrics for comparing nondominated sets," in *Proc. 2002 Congr. Evol. Comput. (CEC)*, vol. 1. Honolulu, HI, USA, pp. 711–716.
[11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[12] A. Farhang-Mehr and S. Azarm, "Diversity assessment of Pareto optimal solution sets: An entropy approach," in *Proc. IEEE Congr. Evol. Comput.*, vol. 1. Honolulu, HI, USA, 2002, pp. 723–728.
[13] K. Deb and S. Jain, "Running performance metrics for evolutionary multi-objective optimization," Dept. Mech. Eng., India Inst. Technol. Kanpur, Kanpur, India, KanGAL, Tech. Rep. 2002004, 2002.
[14] K. Deb, M. Mohan, and S. Mishra, "A fast multi-objective evolutionary algorithm for finding well-spread Pareto-optimal solutions," Dept. Mech. Eng., India Inst. Technol. Kanpur, Kanpur, India, KanGAL, Tech. Rep. 2003002, 2003.
[15] S. Mostaghim and J. Teich, "Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)," in *Proc. 2003 IEEE Swarm Intell. Symp. (SIS)*, pp. 26–33.
[16] S. Mostaghim and J. Teich, "A new approach on many objective diversity measurement," in *Practical Approaches to Multi-Objective Optimization*, J. Branke, K. Deb, K. Miettinen, and R. E. Steuer, Eds. Schloss Dagstuhl, Wadern, Germany: Internat. Begegnungs- und Forschungszentrum für Informatik, 2005.
[17] E. Zitzler, M. Laumanns, L. Thiele, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.
[18] M. Li, J. Zheng, and G. Xiao, "Uniformity assessment for evolutionary multi-objective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Hong Kong, 2008, pp. 625–632.
[19] K. Wang, J. Zheng, and J. Zou, "Neighbor-distance based diversity assessment for multi-objective optimizations," in *Proc. 8th Int. Conf. Natural Comput (ICNC)*, Chongqing, China, May 2012, pp. 833–837.
[20] Z. He and G. G. Yen, "An ensemble method for performance metrics in multiobjective evolutionary algorithms," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, New Orleans, LA, USA, Jun. 2011, pp. 1724–1729.
[21] V. L. S. Silva, E. F. Wanner, S. A. A. G. Cerqueira, and R. H. C. Takahashi, "A new performance metric for multiobjective optimization: The integrated sphere counting," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Singapore, Sep. 2007, pp. 3625–3630.
[22] O. Schutze, X. Esquivel, A. Lara, and C. Coello, "Using the averaged Hausdorff distance as a performance measure in evolutionary multi-objective optimization," *IEEE Trans Evol. Comput.*, vol. 16, no. 4, pp. 504–522, Aug. 2012.
[23] M. Villalobos-Arias, G. Pulido, and C. Coello, "A proposal to use stripes to maintain diversity in a multi-objective particle swarm optimizer," in *Proc. 2005 IEEE Swarm Intell. Symp. (SIS)*, Pasadena, CA, USA, pp. 22–29.

[24] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms—A comparative case study," in *Parallel Problem Solving from Nature—PPSN V* (Lecture Notes in Computer Science), vol. 1498. Berlin, Germany: Springer, 1998, ch. 29, pp. 292–301.

[25] D. Brockhoff, T. Friedrich, and F. Neumann, "Analyzing hypervolume indicator based algorithms," in *Parallel Problem Solving from Nature–PPSN X*. Berlin, Germany: Springer, 2008, pp. 651–660.

[26] E. Zitzler and S. Kunzli, *Indicator-Based Selection in Multiobjective Search* (Lecture Notes in Computer Science), vol. 3242. Berlin, Germany: Springer, 2004, ch. 84, pp. 832–842.

[27] M. Emmerich, N. Beume, and B. Naujoks, *An EMO Algorithm Using the Hypervolume Measure as Selection Criterion* (Lecture Notes in Computer Science), vol. 3410. Berlin, Germany: Springer, 2005, ch. 5, pp. 62–76.

[28] E. Zitzler, D. Brockhoff, and L. Thiele, *The Hypervolume Indicator Revisited: On the Design of Pareto-Compliant Indicators Via Weighted Integration* (Lecture Notes in Computer Science), vol. 4403. Berlin, Germany: Springer, 2007, ch. 64, pp. 862–876.

[29] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, "Theory of the hypervolume indicator, optimal μ-distributions and the choice of the reference point," in *Proc. 10th ACM SIGEVO Workshop Found. Genet. Algorithms*, 2009, p. 87.

[30] N. Beume, C. Fonseca, M. Lopez-Ibanez, L. Paquete, and J. Vahrenhold, "On the complexity of computing the hypervolume indicator," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1075–1082, Oct. 2009.

[31] T. Friedrich, C. Horoba, and F. Neumann, "Multiplicative approximations and the hypervolume indicator," in *Proc. 11th Annu. Conf. Genet. Evol. Comput.*, 2009, pp. 571–578.

[32] D. Brockhoff and E. Zitzler, "Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods," in *Proc. IEEE Congr. Evol. Comput.*, Singapore, Sep. 2007, pp. 2086–2093.

[33] Q. Yang and S. Ding, *Novel Algorithm to Calculate Hypervolume Indicator of Pareto Approximation Set* (Communications in Computer and Information Science), vol. 2. Berlin, Germany: Springer, 2007, ch. 27, pp. 235–244.

[34] T. Ulrich, J. Bader, and E. Zitzler, "Integrating decision space diversity into hypervolume-based multiobjective search," in *Proc. 12th Annu. Conf. Genet. Evol. Comput.*, Portland, OR, USA, 2010, p. 455.

[35] T. Voß, T. Friedrich, K. Bringmann, and C. Igel, "Scaling up indicator-based MOEAs by approximating the least hypervolume contributor: A preliminary study," in *Proc. GECCO Workshop Theor. Aspects Evol. Multiobjective Optim.*, 2010, pp. 1975–1978.

[36] C. Igel, N. Hansen, and S. Roth, "Covariance matrix adaptation for multi-objective optimization," *Evol. Comput.*, vol. 15, no. 1, pp. 1–28, Mar. 2007.

[37] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.

[38] D. W. Corne, J. D. Knowles, and M. J. Oates, *The Pareto Envelope-Based Selection Algorithm for Multiobjective Optimization* (Lecture Notes in Computer Science), vol. 1917. Berlin, Germany: Springer, 2000, ch. 82, pp. 839–848.

[39] I. Karahan and M. Koksalan, "A territory defining multiobjective evolutionary algorithms and preference incorporation," *IEEE Trans. Evol. Comput.*, vol. 14, no. 4, pp. 636–664, Aug. 2010.

[40] J. Knowles and D. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evol. Comput. J.*, vol. 8, no. 2, pp. 149–172, 2000.

[41] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multi-objective optimization," *Evol. Comput.*, vol. 10, no. 3, pp. 263–282, 2002.

[42] M. Li, J. Zheng, R. Shen, K. Li, and Q. Yuan, "A grid-based fitness strategy for evolutionary many-objective optimization," in *Proc. 12th Annu. Conf. Genet. Evol. Comput.*, Portland, OR, USA, 2010, p. 463.

[43] A. Farhang-Mehr and S. Azarm, "Entropy-based multi-objective genetic algorithm for design optimization," *Struct. Multidiscip. Optim.*, vol. 24, no. 5, pp. 351–361, Nov. 2002.

[44] K. Tan, C. Goh, A. Mamun, and E. Ei, "An evolutionary artificial immune system for multi-objective optimization," *Eur. J. Oper. Res.*, vol. 187, no. 2, pp. 371–392, Jun. 2008.

[45] W. LinLin and C. Yunfang, "Diversity based on entropy: A novel evaluation criterion in multi-objective optimization algorithm," *Int. J. Intell. Syst. Appl.*, vol. 4, no. 10, p. 113, Sep. 2012.

[46] J. D. L. Silva and E. K. Burke, "Using diversity to guide the search in multi-objective optimization," in *Applications of Multi-Objective Evolutionary Algorithms* (Advances in Natural Computation), vol. 1. Singapore: World Scientific, Dec. 2004, pp. 727–751.

[47] C. K. Chow and S. Y. Yuen, "A multiobjective evolutionary algorithm that diversifies population by its density," *IEEE Trans. Evol. Comput.*, vol. 16, no. 2, pp. 149–172, Apr. 2012.

[48] S. B. Gee, X. Qiu, and K. C. Tan, "A novel diversity maintenance scheme for evolutionary multi-objective optimization," in *Intelligent Data Engineering and Automated Learning—IDEAL 2013*. Berlin, Germany: Springer, pp. 270–277.

[49] Y. Jin and B. Sendhoff, "A systems approach to evolutionary multiobjective structural optimization and beyond," *IEEE Comput. Intell. Mag.*, vol. 4, no. 3, pp. 62–76, Aug. 2009.

[50] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[51] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

[52] J. Dennis, D. Gay, and R. Welsch, "An adaptive nonlinear least-squares algorithm," *ACM Trans. Math. Softw.*, vol. 7, no. 3, pp. 348–368, Sep. 1981.

[53] K. Deb and R. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, no. 2, pp. 1–34, 1994.

[54] K. Deb and H.-G. Beyer, "Self-adaptive genetic algorithms with simulated binary crossover," *Evol. Comput.*, vol. 9, no. 2, pp. 197–221, Jun. 2001.

[55] K. Deb, K. Sindhya, and T. Okabe, "Self-adaptive simulated binary crossover for real-parameter optimization," in *Proc. Genet. Evol. Comput. Conf. (GECCO)*, 2007, pp. 1187–1194.

[56] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.

[57] Q. Zhang *et al.*, "Multiobjective optimization test instances for the CEC 2009 special session and competition," School Comput. Sci. Elect. Eng., Univ. Essex, Colchester, U.K., Tech. Rep. CES-887, 2008.

[58] C. A. C. Coello and N. C. Cortes, "Solving multiobjective optimization problems using an artificial immune system," *Genet. Program. Evolv. Mach.*, vol. 6, pp. 163–190, Jun. 2005.

[59] E. Ziztler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design, Optimization and Control*. Barcelona, Spain: CIMNE, 2002, pp. 95–100.

[60] C. Goh, K. Tan, D. Liu, and S. Chiam, "A competitive and cooperative co-evolutionary approach to multi-objective particle swarm optimization algorithm design," *Eur. J. Oper. Res.*, vol. 202, no. 1, pp. 42–54, Apr. 2010.

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

[62] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 474–494, Aug. 2013.

Sen Bong Gee received the B.Eng. (Hons.) degree in electrical and computer engineering from National University of Singapore, Singapore, in 2011, where he is currently working toward the Ph.D. degree from the Department of Electrical and Computer Engineering.

His research interests include evolutionary computation, specifically in the dynamic multiobjective optimization.

Kay Chen Tan (SM'08–F'14) received the B.Eng. (Hons.) degree in electronics and electrical engineering and the Ph.D. degree from University of Glasgow, Glasgow, U.K., in 1994 and 1997, respectively.

He is an Associate Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. His research interests include computational and artificial intelligence, with applications to multiobjective optimization, scheduling, automation, data mining, and games. He has published over 100 journal papers and over 100 papers in conference proceedings, and also co-authored five books.

Dr. Tan received the 2012 IEEE Computational Intelligence Society Outstanding Early Career Award for his contributions to evolutionary computation in multiobjective optimization. He was an Editor-in-Chief of *IEEE Computational Intelligence Magazine*. He also is an Associate Editor/Editorial Board member of over 15 international journals.

Vui Ann Shim received the Ph.D. degree in electrical engineering from National University of Singapore, Singapore, in 2012.

He is a Scientist with the Robotics Program, Institute for Infocomm Research, A*STAR, Singapore. His research interests include computational intelligence, computational neuro-science, multiobjective optimization, and robotics.

Nikhil R. Pal (SM'00–F'05) received the B.Sc. (Hons) degree in physics and the M.B.M. degree in operations research from the University of Calcutta, Calcutta, India, in 1979 and 1982, respectively, and the M.Tech and Ph.D. degrees in computer science from the Indian Statistical Institute, Calcutta, India, in 1984 and 1991, respectively.

He is a Professor with the Electronics and Communication Sciences Unit, Indian Statistical Institute, Calcutta, India. His research interests include bioinformatics, brain science, fuzzy logic, image and pattern analysis, neural networks, and evolutionary computation.

Dr. Pal was an Editor-in-Chief of IEEE TRANSACTIONS ON FUZZY SYSTEMS from 2005 to 2010. He is on or has been on the editorial/advisory board/steering committee of several journals. He was a Distinguished Lecturer of the IEEE Computational Intelligence Society (CIS) and was a member of the Administrative Committee of the IEEE CIS from 2010 to 2012. He is currently the Vice President for Publications of the IEEE CIS. He is a fellow of the National Academy of Sciences, India; the Indian National Academy of Engineering; the Indian National Science Academy, and the International Fuzzy Systems Association.