

# PIBEA: Prospect Indicator Based Evolutionary Algorithm for Multiobjective Optimization Problems

Pruet Boonma

Department of Computer Engineering  
Chiang Mai University  
Chiang Mai, 50200, Thailand  
Email: pruet@eng.cmu.ac.th

Junichi Suzuki

Department of Computer Science  
University of Massachusetts, Boston  
Boston, MA 02125, USA  
Email: jxs@cs.umb.edu

**Abstract**—This paper proposes and evaluates an evolutionary multiobjective optimization algorithm (EMOA) that uses a new quality indicator, called the prospect indicator, for parent selection and environmental selection operators. The prospect indicator measures the potential of each individual to reproduce offspring that dominate itself and spread out in the objective space. The prospect indicator allows the proposed EMOA, PIBEA (Prospect Indicator Based Evolutionary Algorithm), to (1) maintain sufficient selection pressure, even in high dimensional MOPs, thereby improving convergence velocity toward the Pareto front, and (2) diversify individuals, even in high dimensional MOPs, thereby distributing individuals uniformly in the objective space. Experimental results show that PIBEA effectively performs its operators in high dimensional problems and outperforms three existing well-known EMOAs, NSGA-II, SPEA2 and AbYSS, in terms of convergence velocity, diversity of individuals, coverage of the Pareto front and performance stability.

## I. INTRODUCTION

This paper proposes and evaluates a new evolutionary algorithm to solve multiobjective optimization problems (MOPs). In general, an MOP is formally described as follows:

$$\left. \begin{array}{l} \text{minimize } F(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})]^T \in \mathcal{O} \\ \text{subject to } \vec{x} = [x_1, x_2, \dots, x_n]^T \in \mathcal{S} \end{array} \right\} \quad (1)$$

$\mathcal{O}$  denotes the objective space.  $\mathcal{S}$  denotes the decision variable space.  $\vec{x} \in \mathcal{S}$  denotes a vector of  $n$  decision variables. It is called an *individual* in evolutionary multiobjective optimization algorithms (EMOAs). A function vector,  $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ , consists of  $m$  real-value objective functions that map an individual ( $\vec{x}$ ) to  $m$  objective values ( $f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})$ ). When  $m \geq 3$ , an MOP is called high-dimensional [1]. The goal of an EMOA is to find an individual(s) that minimize(s) objective values.

In an MOP, there rarely exists a single solution that is optimum with respect to all objectives because objectives often conflict with each other. Therefore, EMOAs often seek the optimal trade-off individuals, or *Pareto-optimal* individuals, by balancing the trade-offs among conflicting objectives. An EMOA evolves a population of individuals via genetic operators (e.g., selection, crossover and mutation operators) through generations toward the *Pareto-optimal front*, or simply *Pareto front*, which is a collection of Pareto-optimal solutions.

The notion of *dominance* plays an important role for EMOAs to seek Pareto optimality. An individual  $\vec{x}$  is said to *dominate* another individual  $\vec{y}$  (denoted by  $\vec{x} \succ \vec{y}$ ) iff  $f_i(\vec{x}) \leq f_i(\vec{y}) \forall i = 1, \dots, m$  and  $f_i(\vec{x}) < f_i(\vec{y}) \exists i = 1, \dots, m$ . EMOAs often rank individuals based on the dominance relationships among them and exploit their ranks in selection operators. This process is called *dominance ranking* [2].

A research trend in the design space of EMOAs is to adopt *indicator-based selection* operators based on performance/quality indicators that augment or replace dominance ranking [3]. For example, the hypervolume indicator have been used in several EMOAs' selection operators [4]–[8].

This paper proposes and evaluates an EMOA that leverages a new quality indicator called *prospect indicator*. The prospect indicator measures the potential of each individual to reproduce offspring that dominate itself and spread out in the objective space. The proposed EMOA, PIBEA (Prospect Indicator Based Evolutionary Algorithm), uses the prospect indicator in its parent selection operator, which chooses individuals from the population to reproduce offspring, as well as its environmental selection operator, which chooses individuals for the next generation from the union of the current-generation individuals and their offspring. The prospect indicator allows the two selection operators to (1) maintain sufficient selection pressure, even in high dimensional MOPs, thereby improving convergence velocity toward the Pareto front, and (2) diversify individuals, even in high dimensional MOPs, thereby distributing individuals uniformly in the objective space.

PIBEA also performs adaptive mutation rate adjustment. It increases the mutation rate to encourage individuals to explore the decision variable space when they have not converged to a part(s) of the Pareto front. However, when they have, it decreases the mutation rate to exploit them and cover a wider range of the front. The proposed operator examines the entropy of individuals to estimate their proximity to the Pareto front.

Experimental results show that PIBEA effectively performs its parent selection, environmental selection and adaptive mutation rate adjustment operators in high dimensional MOPs and outperforms three existing well-known EMOAs, NSGA-II [9], SPEA2 [10] and AbYSS [11], in terms of convergence velocity to the Pareto front, diversity of individuals, coverage of the Pareto front and performance stability.

## II. RELATED WORK

The dominance ranking operator was first proposed in NSGA [2]. NSGA-II extends NSGA with faster dominance ranking, diversity preservation using crowding distance and a  $(\mu + \lambda)$ -elitism in environmental selection [12]. Crowding distance measures the density of each individual and its neighbors in the objective space by computing the Euclidean distances between the individual and its direct neighbors on an objective by objective basis. PIBEA is similar to NSGA-II in that it follows NSGA-II's algorithmic structure (See Section III-A.); however, it extends NSGA-II's selection operators with the prospect indicator. NSGA-II does not consider adaptive mutation rate adjustment as PIBEA does.

SPEA enhances NSGA-II's selection operators with a new fitness assignment method and the notion of elite archiving [13]. SPEA2 further extends SPEA's selection operators with an improved fitness assignment method [10]. It also introduces a new diversity preservation method. It examines the density of each individual and its neighbors in the objective space with a  $k$ -nearest neighbor algorithm. PIBEA is similar to SPEA and SPEA2 in that it provides enhanced selection operators atop NSGA-II's dominance ranking. However, its enhancement strategy is very different from SPEA's and SPEA2's strategies. PIBEA uses the prospect indicator for both parent selection and diversity preservation while SPEA/SPEA2 uses two different methods for parent selection and diversity preservation. SPEA and SPEA2 do not consider adaptive mutation rate adjustment.

Adaptive mutation adjustment was first proposed in [14], and it has been used in [15], [16], among others. In [14], [15], the mutation rate is adjusted based on the fitness values of the current-generation individuals. In [16], the mutation rate is adjusted based on a feedback from the previous generation. For instance, if mutation has limited effects on convergence in the previous generation, the mutation rate is increased by a fixed amount. PIBEA uses an entropy-based diversity metric to decide whether and how much the mutation rate is adjusted.

The hypervolume indicator [4]–[8] is similar to the prospect indicator in that both are volume-based indicators. It measures the volume of a hypercube that each individual dominates in the objective space. The hypercube is formed with the individual and a reference point representing the highest (or worst) possible objective values. In contrast, the prospect indicator measures the volume of a hypercube in the opposite way. It considers a hypercube that dominates an individual. The hypercube is formed with the individual and the Utopian point, which represents the lowest (or best) possible objective values. The Utopian point is the  $(0, 0)$  point in a two dimensional objective space. While the hypervolume indicator requires a carefully chosen reference point depending on an MOP to solve [8], [17], it is always trivial to choose the Utopian point for the prospect indicator regardless of an MOP to solve.

## III. PIBEA

This section describes PIBEA's algorithmic structure (Section III-A) and its operators (Sections III-B to III-D).

### A. Algorithmic Structure

Algorithm 1 shows PIBEA's algorithmic structure, which extends NSGA-II's.

---

#### Algorithm 1 The Algorithmic Structure of PIBEA

---

```

1:  $g = 0$ 
2:  $\mathcal{P}_g = \text{initializePopulation}(N)$ 
3:  $P_m = P_{m_{\max}}$ 
4: while  $g < \text{MAX-GENERATION}$  do
5:    $\mathcal{O}_g = \emptyset$ 
6:   while  $|\mathcal{O}_g| < N$  do do
7:      $p_1 = \text{prospectBasedParentSelection}(\mathcal{P}_g)$ 
8:      $p_2 = \text{prospectBasedParentSelection}(\mathcal{P}_g)$ 
9:     if  $\text{random}() \leq P_c$  then
10:       $\{o_1, o_2\} = \text{crossover}(p_1, p_2)$ 
11:      if  $\text{random}() \leq P_m$  then
12:         $o_1 = \text{mutation}(o_1)$ 
13:      end if
14:      if  $\text{random}() \leq P_m$  then
15:         $o_2 = \text{mutation}(o_2)$ 
16:      end if
17:       $\mathcal{O}_g = \{o_1, o_2\} \cup \mathcal{O}_g$ 
18:    end if
19:  end while
20:  $\mathcal{R}_g = \mathcal{P}_g \cup \mathcal{O}_g$ 
21:  $\mathcal{P}_{g+1} = \text{prospectBasedEnvironmentalSelection}(\mathcal{R}_g)$ 
22:  $\mathcal{E} = \text{computeEntropy}(\mathcal{P}_{g+1})$ 
23:  $P_m = \text{updateMutationRate}(\mathcal{E})$ 
24:  $g = g + 1$ 
25: end while

```

---

In the  $0$ -th generation,  $N$  individuals are randomly generated as the initial population (Line 2). The mutation rate ( $P_m$ ) is initially set to be the maximum value ( $P_{m_{\max}}$ ). In each generation ( $g$ ), a pair of individuals, called parents ( $p_1$  and  $p_2$ ), are chosen from the current population with the proposed parent selection operator, which uses the prospect indicator (`prospectBasedParentSelection()`, Lines 7-8).

With the crossover rate  $P_c$ , two parents reproduce two offspring with the SBX (self-adaptive simulated binary crossover) operator [18] (Lines 10). Each offspring performs polynomial mutation [9] with the probability  $P_m$  (Lines 11 to 16). The selection, crossover and mutation operations are performed repeatedly on  $\mathcal{P}_g$  to produce  $N$  offspring. The offspring ( $\mathcal{O}_g$ ) are combined with the population  $\mathcal{P}_g$  to form  $\mathcal{R}_g$ , which is a pool of candidates for the next-generation individuals.

Environmental selection follows reproduction.  $N$  individuals are selected from  $2N$  individuals in  $\mathcal{R}_g$  as the next-generation population  $\mathcal{P}_{g+1}$  (`prospectBasedEnvironmentalSelection()`, Line 21). Environmental selection performs a  $(N + N)$ -elitism.

Finally, the mutation rate is updated by computing the entropy of the next-generation individuals (Lines 22 and 23).

### B. Prospect Indicator Based Parent Selection

Algorithm 2 shows how the proposed parent selection operator (`prospectBaseParentSelection()` in Algorithm 1) works with the prospect indicator. It is designed as a variant of

binary tournament selection. It randomly draws two individuals from the current population  $\mathcal{P}$  (Lines 1 and 2), compares them based on the dominance relationship between them and chooses a superior one as a parent (Lines 5 to 8). Note that  $p_1 \succ p_2$  means  $p_1$  dominates  $p_2$  as described in Section I

If two individuals ( $p_1$  and  $p_2$ ) do not dominate each other and are placed at the same rank, the proposed operator chooses one of them as a parent with the prospect indicator. Lines 10 and 11 compute the prospect indicator values of  $p_1$  and  $p_2$  ( $I_P(p_1)$  and  $I_P(p_2)$ ), and Line 12 compares the two values. The proposed operator chooses the one with a higher  $I_P$  (Lines 12 to 16).

The prospect indicator value of individual  $i$  ( $I_P(i)$ ) is computed as follows:

$$I_P(i) = V(\mathcal{R}_{rank(i)}) - V(\mathcal{R}_{rank(i)} \setminus \{i\}) \quad (2)$$

$rank(i)$  denotes the value of a rank that  $i$  is placed at.  $\mathcal{R}_{rank(i)}$  denotes a set of individuals that are placed at the same rank as  $i$ .  $V(\mathcal{R})$  denotes the volume of a hypercube that dominates the individuals in  $\mathcal{R}$  in the objective space. It is calculated with the Lebesgue measure as follows.

$$V(\mathcal{R}) = \Lambda \left( \bigcup_{x \in \mathcal{R}} \{x' | x_u \succ x' \succ x\} \right) \quad (3)$$

$x_u$  denotes the Utopian point, and  $\Lambda$  denotes the Lebesgue measure.

The prospect indicator evaluates the potential of an individual to reproduce offspring that dominate itself. Figure 1 shows an example measurement of the prospect indicator in a two-dimensional objective space. This example considers three non-dominated individuals:  $a$ ,  $b$  and  $c$  ( $\mathcal{R}_{rank(a)} = \mathcal{R}_{rank(b)} = \mathcal{R}_{rank(c)} = \{a, b, c\}$ ). The Utopian point is  $(0, 0)$ .  $I_P(b)$  is a shaded area (i.e.,  $V(\mathcal{R}_{rank(b)}) - V(\mathcal{R}_{rank(b)} \setminus \{b\})$ ). It is the area where individual  $b$  can exclusively reproduce its offspring by evolving itself.

---

#### Algorithm 2 prospectBaseParentSelection()

---

**Require:**  $\mathcal{P} | \mathcal{P} \neq \emptyset$

- 1:  $p_1 = \text{randomSelection}(\mathcal{P})$
- 2:  $p_2 = \text{randomSelection}(\mathcal{P})$
- 3: **if**  $p_1 = p_2$  **then**
- 4:     **return**  $p_1$
- 5: **else if**  $p_1 \succ p_2$  **then**
- 6:     **return**  $p_1$
- 7: **else if**  $p_2 \succ p_1$  **then**
- 8:     **return**  $p_2$
- 9: **else**
- 10:     $I_P(p_1) = \text{prospectIndicator}(p_1, \mathcal{R}_{rank(p_1)})$
- 11:     $I_P(p_2) = \text{prospectIndicator}(p_2, \mathcal{R}_{rank(p_2)})$
- 12:    **if**  $I_P(p_1) > I_P(p_2)$  **then**
- 13:     **return**  $p_1$
- 14:    **else**
- 15:     **return**  $p_2$
- 16:    **end if**
- 17: **end if**

---



---

#### Algorithm 3 prospectIndicator()

---

**Require:**  $p, \mathcal{P} | \mathcal{P} \neq \emptyset$

- 1:  $v = 1$
- 2: **for each**  $o \in \mathcal{O}$  **do**
- 3:     $s = \emptyset$
- 4:    **for each**  $n \in \mathcal{P}$  **do**
- 5:     **if**  $f_o(n) < f_o(p)$  **then**
- 6:      **if**  $s = \emptyset$  **then**
- 7:         $s = n$
- 8:      **else if**  $f_o(s) < f_o(n)$  **then**
- 9:         $s = n$
- 10:     **end if**
- 11:    **end if**
- 12: **end for**
- 13:  $v = v \times |f_o(p) - f_o(s)|$
- 14: **end for**
- 15: **return**  $v$

---

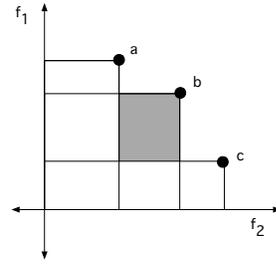


Fig. 1: An Example Measurement of Prospect Indicator

Algorithm 3 shows pseudo code to compute  $I_P(p)$ .  $\mathcal{P}$  denotes a set of individuals that are placed at the same rank as individual  $p$ . For each objective ( $o$ ), the distance between  $p$  and  $s$  is measured to compute  $I_P(p)$ , where  $s$  denotes an individual that yields the closest yet superior objective value.

#### C. Prospect Indicator Based Environmental Selection

Algorithm 4 shows how the proposed environmental selection operator (`prospectBaseEnvironmentalSelection()` in Algorithm 1) works with the prospect indicator. In the environmental selection process,  $N$  individuals are selected from  $2N$  individuals in  $\mathcal{R}_g$  as the next-generation population ( $\mathcal{P}_{g+1}$ ). The individuals in  $\mathcal{R}_g$  are selected and moved to  $\mathcal{P}_{g+1}$  on a rank by rank basis, starting with the top rank (Line 2 to 8). In Line 3, `nonDominatedIndividualSelection()` determines the dominance relationships among a given set of individuals and chooses the non-dominated (i.e., top-ranked) ones as  $\mathcal{R}_{ND}$ . Algorithm 5 shows pseudo code of this operator.

If the number of individuals in  $\mathcal{P}_{g+1} \cup \mathcal{R}_{ND}$  is less than  $N$ ,  $\mathcal{R}_{ND}$  moves to  $\mathcal{P}_{g+1}$ . Otherwise, a subset of  $\mathcal{R}_{ND}$  moves to  $\mathcal{P}_{g+1}$ . The subset is selected with `subsetSelection()` (Line 5). Algorithm 6 shows how this operator works with the prospect indicator. It sorts the individuals in  $\mathcal{R}_{ND}$  from the ones with higher  $I_P$  values to the ones with lower  $I_P$  values (Line 4). The individuals with higher  $I_P$  values have higher chances to be selected to  $\mathcal{P}_{g+1}$  (Lines 6 to 9)

The prospect indicator indicates how distant each individual is to its direct neighbors in the objective space. For example, in Figure 1,  $I_P(b)$  indicates how distant individual  $b$  is to

individuals  $a$  and  $c$ . A higher  $I_P(b)$  means that  $b$  is more distant to  $a$  and  $c$ , and a lower  $I_P(b)$  means that  $b$  is closer to  $a$  and  $c$ . Therefore, the prospect indicator evaluates the potential of an individual to reproduce offspring that spread out in the objective space and contribute to the population's diversity.

---

**Algorithm 4** prospectBasedEnvironmentalSelection()

---

**Require:**  $\mathcal{R}_g | \mathcal{R}_g \neq \emptyset$   
1:  $\mathcal{P}_{g+1} = \emptyset$   
2: **while**  $|\mathcal{P}_{g+1}| < N$  **do**  
3:  $\mathcal{R}_{ND} = \text{nonDominatedIndividualSelection}(\mathcal{R}_g \setminus \mathcal{P}_{g+1})$   
4: **if**  $|\mathcal{P}_{g+1}| + |\mathcal{R}_{ND}| > N$  **then**  
5:  $\mathcal{R}_{ND} = \text{subsetSelection}(\mathcal{R}_{ND}, N - |\mathcal{P}_{g+1}|)$   
6: **end if**  
7:  $\mathcal{P}_{g+1} = \mathcal{P}_{g+1} \cup \mathcal{R}_{ND}$   
8: **end while**  
9: **return**  $\mathcal{P}_{g+1}$

---



---

**Algorithm 5** nonDominatedIndividualSelection()

---

**Require:**  $\mathcal{P} | \mathcal{P} \neq \emptyset$   
1:  $\mathcal{P}' = \emptyset$   
2: **for each**  $p \in \mathcal{P} \wedge p \notin \mathcal{P}'$  **do**  
3:  $\mathcal{P}' = \mathcal{P}' \cup \{p\}$   
4: **for each**  $q \in \mathcal{P}' \wedge q \neq p$  **do**  
5: **if**  $p \succ q$  **then**  
6:  $\mathcal{P}' = \mathcal{P}' \setminus \{q\}$   
7: **else if**  $q \succ p$  **then**  
8:  $\mathcal{P}' = \mathcal{P}' \setminus \{p\}$   
9: **end if**  
10: **end for**  
11: **end for**  
12: **return**  $\mathcal{P}'$

---



---

**Algorithm 6** subsetSelection()

---

**Require:**  $\mathcal{P}, T | \mathcal{P} \neq \emptyset$   
1: **for each**  $p \in \mathcal{P}$  **do**  
2:  $I_P(p) = \text{prospectIndicator}(p, \mathcal{P})$   
3: **end for**  
4:  $\mathcal{P}_S = \text{sort}(\mathcal{P}, I_P)$   
5:  $\mathcal{R} = \emptyset$   
6: **for each**  $p \in \mathcal{P}_S \wedge |\mathcal{R}| < T$  **do**  
7:  $\mathcal{R} = \mathcal{R} \cup p$   
8: **end for**  
9: **return**  $\mathcal{R}$

---

#### D. Adaptive Mutation Rate Adjustment

The proposed mutation rate adjustment operator is designed to balance exploration and exploitation in an optimization process. It increases the mutation rate ( $p_m$  in Algorithm 1) to encourage individuals to explore the decision variable space when they have not converged enough toward the Pareto front. This aims to increase convergence velocity by placing a higher priority on exploration. On the contrary, when individuals have not converged enough to the Pareto front, the proposed operator decreases the mutation rate to exploit them and cover a wider range of the front. This aims to improve the

diversity of individuals and stabilize convergence with limited performance fluctuations across generations.

The proposed operator examines the entropy of individuals in the objective space in order to estimate their proximity to the Pareto front. It decreases the mutation rate when the entropy increases based on the observation that randomly-generated individuals are disordered in the objective space at the first generation and they become more ordered through generations.

The entropy of individuals is computed with hypercubes that comprise the objective space. As an example in Figure 2 shows, the objective space is bounded by the maximum and minimum objective values that individuals yield. Then, it is divided to hypercubes. Figure 2 shows six individuals in the three dimensional objective space that consists of eight hypercubes.

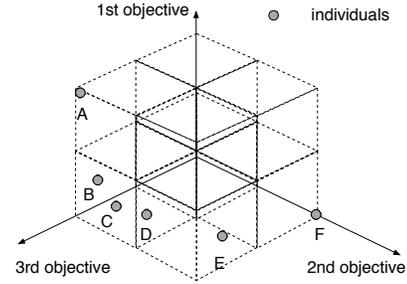


Fig. 2: An Example Hypercube

The entropy of individuals ( $H$ ) is computed as:

$$H = - \sum_{i \in C} P(i) \log_2(P(i)) \quad (4)$$

where

$$P(i) = \frac{n_i}{\sum_{i \in C} n_i}. \quad (5)$$

$C$  denotes the set of hypercubes in the objective space.  $P(i)$  denotes the probability that individuals exist in a hypercube  $i$ .  $n_i$  denotes the number of individuals in a hypercube  $i$ . In Figure 2,  $\sum_{i \in C} n_i = 6$  and  $C = 8$ .

Once entropy ( $H$ ) is obtained, it is normalized as follows:

$$H_o = \frac{H}{H_{\max}} = \frac{H}{\log_2 |C|} \quad (6)$$

The mutation rate  $m$  is adjusted with  $H_o$  as follow.

$$P_m = P_{m_{\max}} \times \sqrt{1 - (1 - H_o)^2} \quad (7)$$

$P_{m_{\max}}$  is the maximum mutation rate.

#### IV. EXPERIMENTAL RESULTS

This section evaluates PIBES with 45 MOPs that are derived from three standard test problems: DTLZ1, DTLZ3 and DTLZ7 [19] with a varying number of objectives ( $m$ , three to nine) and decision variables ( $n$ , 12 to 24). This section also compares PIBES with three existing EMOAs: NSGA-II, SPEA2 and AbYSS. AbYSS is designed to be a best-of-breed

algorithm that integrates the operators in NSGAII, SPEA2 and PAES [20]: crowding distance in NSGA-II, the initial individual selection with density estimation in SPEA2 and elite archiving in PAES. AbYSS also incorporates the scatter search template [21], which enables scatter search for the local search and diversification purposes.

TABLE I: Parameter Setup

Parameter	value
Maximum Number of Generations	250
Population Size	100
SPEA2 and AbYSS External Archive Size	100
AbYSS Reference Sets Size	10/10
AbYSS Improvement Method Iteration	5
NSGA-II's and SPEA2's Crossover Rate	0.9
Crossover Rate ( $P_c$ )	0.9
Crossover Operator	SBX
Crossover Distribution Index ( $\eta_c$ )	20
NSGA-II's and SPEA2's Mutation Rate	1/(# of decision variables)
PIBEA's Maximum Mutation Rate ( $m_{max}$ )	1/(# of decision variables)
Mutation Operator	Polynomial Mutation
Mutation Distribution Index ( $\eta_m$ )	20
Size of $V_{GD}$ moving window ( $w$ )	10

NSGA-II, SPEA2 and AbYSS were configured as described in [9], [10] and [11], respectively. All experiments were conducted with jMetal [22]. Each experimental result is the average of 20 independent results.

#### A. Performance Metrics

This paper uses the following four evaluation metrics.

- *Generational distance* (GD): measures the average distance from non-dominated individuals toward the Pareto-optimal front [23].
- *Extend* (EX): measures the coverage of non-dominated individuals in the objective space [9].
- *Spread* (SP): measures the distribution of non-dominated individuals in the objective space [11].
- *Variance of Generational distance* ( $V_{GD}$ ): is the variance of GD values in the last  $w$  generations (Table I). It measures the stability of GD performance that non-dominated individuals yield. A large  $V_{GD}$  indicates greater fluctuation in the GD performance

#### B. Convergence

This section discusses the convergence of individuals with the GD metric. Table II shows the results from DTLZ1 problem after 250 generations. The lower GD value indicate that the non-dominated individuals are closer to the Pareto-optimal front; thus, lower is better. The first and second columns of the tables show the number of objectives ( $m$ ) and decision variables ( $n$ ) used in experiments, respectively. Bold number represents the best results among four algorithms. The result of GD shows that non-dominated individuals from PIBEA are closer to Pareto-optimal front than that from the other algorithms in every combination of number of objectives and decision variables. Prospect based parent selection and prospect based environmental selection allow PIBEA to find better non-dominated individuals, in term of closeness to Pareto-optimal front.

TABLE II: GD Results with DTLZ1

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>5.3355E-02</b>	9.0700E-01	9.8983E-02	3.0058E+00
	15	<b>6.3209E-02</b>	4.8432E-01	4.0401E-01	4.3580E+00
	18	<b>8.0557E-02</b>	3.3287E-01	8.4416E-01	5.1871E+00
	21	<b>1.7821E-01</b>	1.3229E+00	1.1248E+00	7.4520E+00
6	12	<b>5.9074E-02</b>	8.9780E+00	8.0647E+00	3.7546E+00
	15	<b>2.5001E-01</b>	1.1386E+01	1.2785E+01	6.2831E+00
	18	<b>3.2716E-01</b>	1.6268E+01	1.4880E+01	7.8906E+00
	21	<b>4.3740E-01</b>	1.7996E+01	2.0922E+01	9.9878E+00
9	12	<b>4.9282E-02</b>	1.0190E+01	1.3020E+01	2.5674E+00
	15	<b>9.1175E-02</b>	1.6761E+01	2.6484E+01	5.5819E+00
	18	<b>2.5214E-01</b>	2.8280E+01	3.7333E+01	8.4386E+00
	21	<b>3.6007E-01</b>	2.8424E+01	3.8749E+01	1.0473E+01
24	<b>1.1466E+00</b>	4.2360E+01	3.7094E+01	1.2496E+01	

TABLE III: GD Results with DTLZ3

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>8.5373E-02</b>	5.1257E-01	1.0775E-01	3.0043E+00
	15	<b>9.7620E-02</b>	1.2090E+00	2.5584E-01	4.2776E+00
	18	<b>1.4062E-01</b>	2.1203E+00	4.3341E-01	1.0132E+01
	21	<b>3.6745E-01</b>	2.4634E+00	6.8791E-01	1.2144E+01
6	12	<b>1.1397E-01</b>	8.2864E+00	7.9889E+00	5.7849E+00
	15	<b>4.3240E-01</b>	1.1728E+01	1.0807E+01	8.5492E+00
	18	<b>3.3532E-01</b>	1.5359E+01	1.4189E+01	1.0961E+01
	21	<b>4.4475E-01</b>	2.0274E+01	1.6232E+01	1.5244E+01
9	12	<b>9.0554E-02</b>	7.8765E+00	7.8911E+00	2.6630E+00
	15	<b>1.0764E-01</b>	1.6385E+01	1.3182E+01	6.0813E+00
	18	<b>3.3427E-01</b>	2.0609E+01	1.7826E+01	1.1743E+01
	21	<b>4.6355E-01</b>	2.6850E+01	2.7159E+01	1.2875E+01
24	<b>7.5107E-01</b>	3.0623E+01	2.7626E+01	1.6836E+01	

Fig. 4a shows the value of GD with respect to the number of generations. In the figure, the number of objectives is three and the number of decision variables is 12. The figure shows that all four algorithms can improve GD and the value of GD to a particular value; however, PIBEA can improve the non-dominated individuals faster than the other algorithms which can be observed from steeper slope in the first fifty generations. NSGA-II and SPEA2 have similar performance while AbYSS is the worse one. Fig. 4b shows the value of GD when the number of objectives is increased to nine and the number of decision variables is increased to 24. In this figure, only PIBEA can improve the value of GD while the other algorithms cannot. PIBEA can find solutions faster for simple problem, i.e., with three objectives, and allow GD to converge in complicated problem, i.e., with nine objectives, while the other cannot.

Table III shows the GD results from DTLZ3 problem. Even though, DTLZ3 problem is harder than DTLZ1; however, the same result as from DTLZ1 can be observed.

Table III shows the GD results from DTLZ7 problem. Even though, DTLZ7 problem is harder than DTLZ1 and DTLZ3; however, the similar result as from DTLZ1 and DTLZ3 can be observed. In particular, PIBEA can find better solutions in most of configurations, except when the number of objectives is 9 and the number of decision variables are 12, 18 and 24.

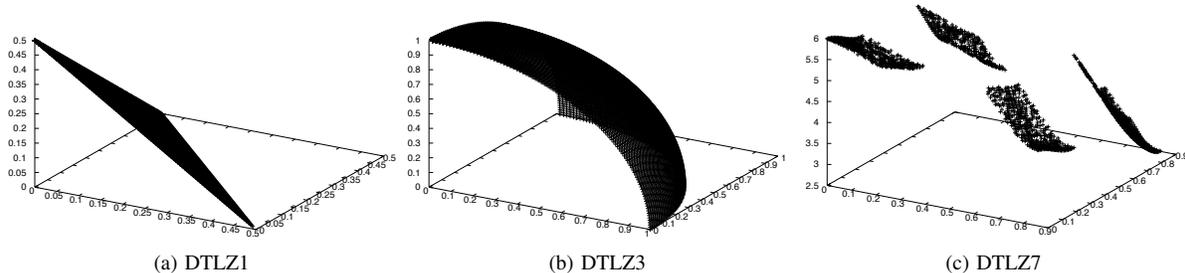
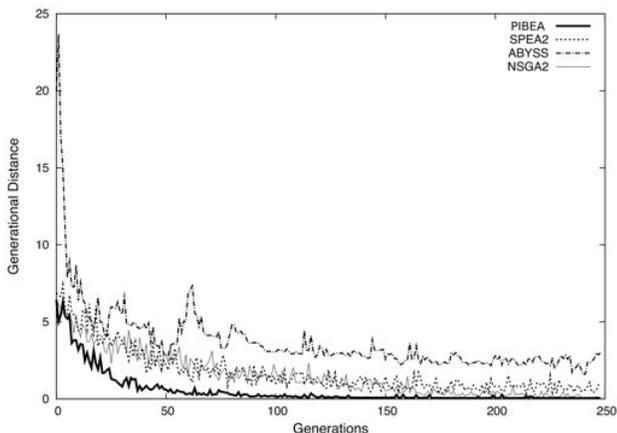
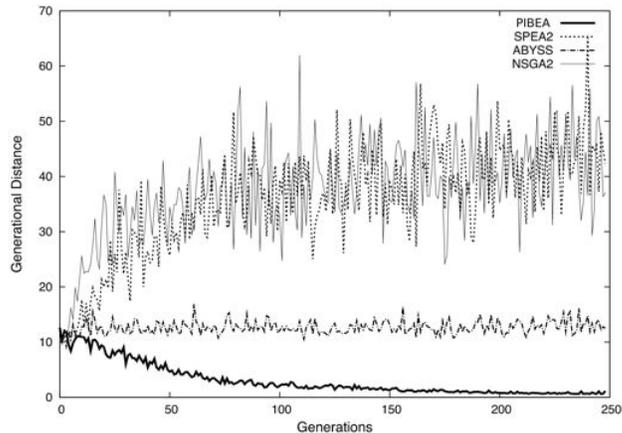


Fig. 3: The Pareto Front Shapes of DTLZ1, DTLZ3 and DTLZ7 with three Objectives



(a) 3 Objectives and 12 Decision Variables



(b) 9 Objectives and 24 Decision Variables

Fig. 4: GD Transition with DTLZ1

TABLE IV: GD Results with DTLZ7

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.3692E-01</b>	1.4364E-01
	15	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.3692E-01</b>	1.4425E-01
	18	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.3692E-01</b>	1.5632E-01
	21	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.3692E-01</b>	1.4994E-01
	24	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.3692E-01</b>	<b>1.5376E-01</b>
6	12	<b>1.2862E-01</b>	1.3081E-01	1.2883E-01	1.3111E-01
	15	<b>1.2862E-01</b>	1.2975E-01	1.2865E-01	1.3244E-01
	18	<b>1.2862E-01</b>	1.2922E-01	1.2864E-01	1.3146E-01
	21	<b>1.2865E-01</b>	1.2898E-01	<b>1.2865E-01</b>	1.3195E-01
	24	<b>1.2865E-01</b>	1.2898E-01	1.2871E-01	1.3172E-01
9	12	1.6165E-01	1.6541E-01	<b>1.6103E-01</b>	1.6461E-01
	15	<b>1.6122E-01</b>	1.6512E-01	1.6216E-01	1.6279E-01
	18	1.6179E-01	1.6488E-01	<b>1.6147E-01</b>	1.6304E-01
	21	<b>1.6162E-01</b>	1.6325E-01	1.6163E-01	1.6400E-01
	24	1.6168E-01	1.6222E-01	<b>1.6125E-01</b>	1.6430E-01

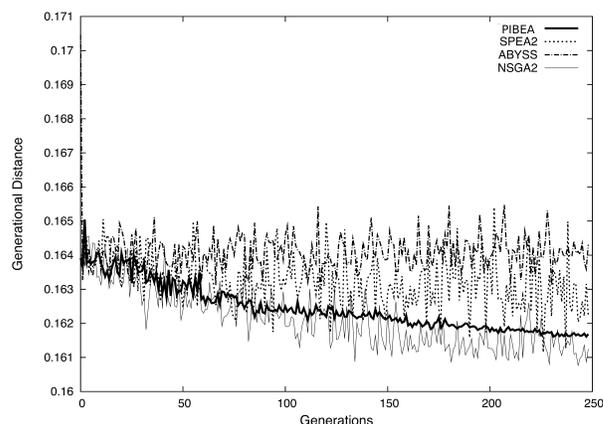


Fig. 5: GD Transition with DTLZ7 (9 Objectives and 24 Decision Variables)

The results in Table IV might suggest that NODAME is worse than NSGA-II when the number of objectives is large. However, Fig. 5, which is GD result from DTLZ7 problem with 9 objectives and 24 decision variables, shows that the value of GD from NSGA-II fluctuate from generation to generation while NODAME's GD is more stable. This is because of adaptive mutation rate adjustment in NODAME. Thus, the results show that even though NODAME does not have better performance in terms of GD at 250th generation,

but it provides more stable solutions than that of NSGA-II. The stability aspect of experimental results will be discussed more in the section IV-D.

### C. Coverage and Distribution

This section discusses the coverage of individuals with the EX metric and the distribution of individuals with the SP

TABLE V: EX Results with DTLZ1

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.8392E+00</b>	2.8274E-02	2.7868E-01	9.3400E-03
	15	<b>6.1102E-01</b>	5.2211E-02	6.9642E-02	6.7836E-03
	18	<b>3.6711E-01</b>	6.4069E-02	3.3673E-02	5.3059E-03
	21	<b>1.4725E-01</b>	1.9471E-02	2.3975E-02	3.2016E-03
	24	<b>8.0374E-02</b>	1.5248E-02	2.4030E-02	2.3813E-03
6	12	<b>1.2746E+00</b>	4.7770E-03	5.7031E-03	1.0022E-02
	15	<b>2.3297E-01</b>	3.4767E-03	3.4841E-03	6.1213E-03
	18	<b>1.6459E-01</b>	2.5507E-03	3.0292E-03	4.8357E-03
	21	<b>1.2088E-01</b>	2.1617E-03	2.3455E-03	3.9077E-03
	24	<b>1.0608E-01</b>	1.8104E-03	2.4236E-03	2.8229E-03
9	12	<b>4.5035E+00</b>	5.3875E-03	5.6042E-03	2.1390E-02
	15	<b>1.0337E+00</b>	3.4898E-03	2.7577E-03	8.3928E-03
	18	<b>3.0551E-01</b>	2.1416E-03	1.9426E-03	5.8827E-03
	21	<b>2.2812E-01</b>	2.2813E-03	1.8143E-03	4.3775E-03
	24	<b>7.3816E-02</b>	1.5133E-03	1.7898E-03	4.1402E-03

TABLE VI: EX Results with DTLZ3

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.8392E+00</b>	2.8274E-02	2.7868E-01	9.3400E-03
	15	<b>6.1102E-01</b>	5.2211E-02	6.9642E-02	6.7836E-03
	18	<b>3.6711E-01</b>	6.4069E-02	3.3673E-02	5.3059E-03
	21	<b>1.4725E-01</b>	1.9471E-02	2.3975E-02	3.2016E-03
	24	<b>8.0374E-02</b>	1.5248E-02	2.4030E-02	2.3813E-03
6	12	<b>1.2746E+00</b>	4.7770E-03	5.7031E-03	1.0022E-02
	15	<b>2.3297E-01</b>	3.4767E-03	3.4841E-03	6.1213E-03
	18	<b>1.6459E-01</b>	2.5507E-03	3.0292E-03	4.8357E-03
	21	<b>1.2088E-01</b>	2.1617E-03	2.3455E-03	3.9077E-03
	24	<b>1.0608E-01</b>	1.8104E-03	2.4236E-03	2.8229E-03
9	12	<b>4.5035E+00</b>	5.3875E-03	5.6042E-03	2.1390E-02
	15	<b>1.0337E+00</b>	3.4898E-03	2.7577E-03	8.3928E-03
	18	<b>3.0551E-01</b>	2.1416E-03	1.9426E-03	5.8827E-03
	21	<b>2.2812E-01</b>	2.2813E-03	1.8143E-03	4.3775E-03
	24	<b>7.3816E-02</b>	1.5133E-03	1.7898E-03	4.1402E-03

TABLE VII: EX Results with DTLZ7

$m$	$n$	NODAME	SPEA2	NSGA-II	AbYSS
3	12	<b>1.6414E+00</b>	1.3690E+00	1.4667E+00	1.2143E+00
	15	<b>1.5777E+00</b>	1.3358E+00	1.3069E+00	1.0802E+00
	18	<b>1.5582E+00</b>	1.3321E+00	1.3680E+00	1.2234E+00
	21	<b>1.5046E+00</b>	1.3906E+00	1.4676E+00	1.0154E+00
	24	<b>1.5510E+00</b>	1.2686E+00	1.3332E+00	9.8572E-01
6	12	<b>3.2083E+00</b>	1.3056E+00	1.6625E+00	1.0589E+00
	15	<b>3.1185E+00</b>	1.2484E+00	1.8346E+00	1.0034E+00
	18	<b>3.3767E+00</b>	1.3658E+00	1.6214E+00	1.0609E+00
	21	<b>3.2537E+00</b>	1.6246E+00	1.6676E+00	9.6770E-01
	24	<b>3.1521E+00</b>	1.6549E+00	1.9483E+00	9.4542E-01
9	12	<b>1.7976E+00</b>	1.1416E+00	1.5542E+00	1.0881E+00
	15	<b>2.0924E+00</b>	1.0412E+00	1.3217E+00	1.1656E+00
	18	<b>1.7067E+00</b>	1.0817E+00	1.4234E+00	1.1435E+00
	21	<b>1.9529E+00</b>	1.1386E+00	1.4216E+00	1.0462E+00
	24	<b>1.6634E+00</b>	1.4051E+00	1.5740E+00	1.0457E+00

metric. Table V, VI and VII show the results of EX metric on DTLZ1, DTLZ3 and DTLZ7 problems, respectively. The higher value indicates better coverage. The results from the tables show that non-dominated individuals from PIBEA have better coverage than that from the other algorithms in all combination of the number of objectives and decision variables. The experimental results point out that prospect based parent selection and prospect based environmental selection in PIBEA allow non-dominated individuals to have better coverage of Pareto-optimal front.

Table VIII, X and IX show the results of SP metric on DTLZ1, DTLZ3 and DTLZ7 problems, respectively. The lower value indicates better distribution. The results from the tables show that non-dominated individuals from PIBEA

TABLE VIII: SP Results with DTLZ1

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.2080E+00</b>	1.3351E+00	1.3331E+00	1.3400E+00
	15	<b>1.2887E+00</b>	1.3361E+00	1.3357E+00	1.3393E+00
	18	<b>1.1676E+00</b>	1.1898E+00	1.1905E+00	1.1915E+00
	21	<b>1.3320E+00</b>	1.3398E+00	1.3398E+00	1.3407E+00
	24	<b>1.3353E+00</b>	1.3404E+00	1.3397E+00	1.3407E+00
6	12	<b>1.2867E+00</b>	1.3407E+00	1.3408E+00	1.3403E+00
	15	<b>1.3211E+00</b>	1.3408E+00	1.3408E+00	1.3406E+00
	18	<b>1.3322E+00</b>	1.3408E+00	1.3408E+00	1.3407E+00
	21	<b>1.3364E+00</b>	1.3408E+00	1.3408E+00	1.3407E+00
	24	<b>1.3365E+00</b>	1.3409E+00	1.3408E+00	1.3408E+00
9	12	<b>1.2485E+00</b>	1.3409E+00	1.3408E+00	1.3406E+00
	15	<b>1.3145E+00</b>	1.3409E+00	1.3409E+00	1.3408E+00
	18	<b>1.3328E+00</b>	1.3409E+00	1.3409E+00	1.3409E+00
	21	<b>1.3378E+00</b>	1.3409E+00	1.3409E+00	1.3409E+00
	24	<b>1.3397E+00</b>	1.3409E+00	1.3409E+00	1.3409E+00

TABLE IX: SP Results with DTLZ3

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.1428E+00</b>	1.2630E+00	1.2549E+00	1.2712E+00
	15	<b>1.3230E+00</b>	1.3387E+00	1.3359E+00	1.3360E+00
	18	<b>1.3237E+00</b>	1.3401E+00	1.3367E+00	1.3376E+00
	21	<b>1.3326E+00</b>	1.3402E+00	1.3391E+00	1.3384E+00
	24	<b>1.3344E+00</b>	1.3394E+00	1.3393E+00	1.3393E+00
6	12	<b>1.2818E+00</b>	1.3407E+00	1.3407E+00	1.3404E+00
	15	<b>1.3234E+00</b>	1.3408E+00	1.3408E+00	1.3407E+00
	18	<b>1.3336E+00</b>	1.3408E+00	1.3408E+00	1.3407E+00
	21	<b>1.3364E+00</b>	1.3408E+00	1.3408E+00	1.3408E+00
	24	<b>1.3389E+00</b>	1.3409E+00	1.3409E+00	1.3408E+00
9	12	<b>1.2160E+00</b>	1.3408E+00	1.3407E+00	1.3405E+00
	15	<b>1.2958E+00</b>	1.3408E+00	1.3408E+00	1.3408E+00
	18	<b>1.3326E+00</b>	1.3409E+00	1.3409E+00	1.3408E+00
	21	<b>1.3377E+00</b>	1.3409E+00	1.3409E+00	1.3408E+00
	24	<b>1.3392E+00</b>	1.3409E+00	1.3409E+00	1.3409E+00

TABLE X: SP Results with DTLZ7

$m$	$n$	NODAME	SPEA2	NSGA-II	AbYSS
3	12	<b>1.2051E+00</b>	1.2315E+00	1.2329E+00	1.2428E+00
	15	<b>1.2089E+00</b>	1.2375E+00	1.2621E+00	1.2547E+00
	18	<b>1.2107E+00</b>	1.2417E+00	1.2432E+00	1.2398E+00
	21	<b>1.2140E+00</b>	1.2354E+00	1.2172E+00	1.2611E+00
	24	<b>1.2103E+00</b>	1.2477E+00	1.2420E+00	1.2632E+00
6	12	<b>1.1983E+00</b>	1.2885E+00	1.2625E+00	1.3103E+00
	15	<b>1.1984E+00</b>	1.2973E+00	1.2424E+00	1.3124E+00
	18	<b>1.2002E+00</b>	1.2788E+00	1.2617E+00	1.3095E+00
	21	<b>1.2135E+00</b>	1.2682E+00	1.2414E+00	1.3137E+00
	24	<b>1.2150E+00</b>	1.2729E+00	1.2528E+00	1.3152E+00
9	12	<b>1.2730E+00</b>	1.3055E+00	1.2850E+00	1.3112E+00
	15	<b>1.2694E+00</b>	1.3134E+00	1.2972E+00	1.3106E+00
	18	<b>1.2829E+00</b>	1.3118E+00	1.2959E+00	1.3094E+00
	21	<b>1.2776E+00</b>	1.3068E+00	1.2954E+00	1.3126E+00
	24	<b>1.2865E+00</b>	<b>1.2895E+00</b>	<b>1.2850E+00</b>	1.3130E+00

have better distribution than that from the other algorithms in all combination of the number of objectives and decision variables.

#### D. Stability

This section describes the stability of PIBEA and other algorithms with the  $V_{GD}$  metric. Tables XI, XII and XIII show the results of  $V_{GD}$  metric on DTLZ1, DTLZ3 and DTLZ7 problems, respectively. The lower value indicated better stability, i.e., less fluctuation of the GD value from generation to generation. The results from the tables show that the Pareto-optimal front from PIBEA are more stable than that from the other algorithms in all combination of the number of objectives and decision variables, except only one configuration in DTLZ7 problem. The experimental results

TABLE XI:  $V_{GD}$  Results with DTLZ1

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>1.2581E-05</b>	2.0156E+00	5.8892E-03	6.3816E+00
	15	<b>2.9353E-04</b>	1.4628E+00	8.6292E-01	1.1681E+01
	18	<b>1.6589E-04</b>	2.9451E+00	1.4882E+00	1.5173E+01
	21	<b>4.0068E-02</b>	7.9902E+00	2.3463E+00	4.7788E+01
	24	<b>2.4267E-02</b>	6.8764E+00	3.5538E+00	2.6704E+01
6	12	<b>2.3797E-02</b>	1.6900E+01	4.3322E+01	8.0781E+00
	15	<b>8.5834E-02</b>	1.4285E+02	1.0470E+02	1.6212E+01
	18	<b>4.3545E-02</b>	1.6356E+02	1.5827E+02	2.3014E+01
	21	<b>1.7039E-01</b>	9.5904E+01	2.0596E+02	2.0612E+01
	24	<b>3.3212E-02</b>	3.6222E+02	2.5297E+02	3.1111E+01
9	12	<b>5.3941E-06</b>	2.6161E+02	1.0868E+02	3.2979E+00
	15	<b>5.0623E-02</b>	7.5924E+02	4.1223E+02	1.9750E+01
	18	<b>3.5187E-01</b>	1.2626E+03	8.8118E+02	3.7077E+01
	21	<b>6.6444E-01</b>	2.8234E+03	2.7988E+03	2.6021E+01
	24	<b>1.0937E+00</b>	2.3589E+03	2.6697E+03	6.0432E+01

TABLE XII:  $V_{GD}$  Results with DTLZ3

$m$	$n$	PIBEA	SPEA2	NSGA-II	AbYSS
3	12	<b>6.6312E-09</b>	2.5692E+00	1.7575E-02	5.8391E-01
	15	<b>2.1694E-01</b>	3.5077E+00	2.8959E-01	9.2322E+00
	18	<b>1.1522E-01</b>	8.0945E+00	2.2607E-01	1.0195E+01
	21	<b>1.1807E-01</b>	1.2602E+01	4.2468E+00	2.6642E+01
	24	<b>3.4678E-01</b>	7.0695E+00	2.9145E+00	2.9250E+01
6	12	<b>1.8787E-03</b>	2.5630E+01	2.1531E+01	6.8258E+00
	15	<b>3.7585E-01</b>	2.1979E+01	1.7941E+01	1.7955E+01
	18	<b>8.4842E-01</b>	2.2775E+01	7.5170E+01	2.4103E+01
	21	<b>4.1256E-01</b>	4.4103E+01	5.9665E+01	6.8925E+01
	24	<b>1.2688E+00</b>	4.3630E+01	9.0292E+01	1.8932E+01
9	12	<b>6.7759E-05</b>	7.7099E+00	4.1550E+01	1.6751E+01
	15	<b>1.8426E-03</b>	3.4533E+01	4.8364E+01	7.4054E+00
	18	<b>2.4059E-01</b>	6.0168E+01	7.9266E+01	3.5756E+01
	21	<b>3.1527E-01</b>	3.3641E+01	3.1145E+02	2.1345E+01
	24	<b>2.0212E+00</b>	5.5618E+01	8.5817E+01	4.0371E+01

TABLE XIII:  $V_{GD}$  Results with DTLZ7

$m$	$n$	NODAME	SPEA2	NSGA-II	AbYSS
3	12	<b>6.5603E-16</b>	1.5021E-11	3.1399E-12	6.8922E-06
	15	<b>1.0236E-13</b>	6.7161E-12	1.4847E-12	1.6725E-05
	18	1.1940E-12	1.2446E-12	<b>3.5992E-13</b>	1.2875E-05
	21	<b>1.0216E-15</b>	1.9177E-12	6.9211E-13	5.8130E-06
	24	<b>1.1220E-15</b>	7.6102E-13	4.5970E-13	9.7151E-06
6	12	<b>4.3418E-11</b>	8.8891E-06	4.1124E-07	2.4329E-05
	15	<b>1.4807E-10</b>	2.8530E-06	2.8575E-07	6.6064E-06
	18	<b>1.2706E-10</b>	1.0514E-06	1.2183E-08	7.4371E-06
	21	<b>1.4458E-09</b>	3.9028E-07	1.2570E-08	8.6652E-06
	24	<b>9.9971E-10</b>	5.4039E-07	4.5637E-08	9.7190E-06
9	12	<b>8.0629E-07</b>	7.2187E-05	4.1225E-05	1.9459E-05
	15	<b>1.8890E-07</b>	5.1641E-05	3.3974E-05	2.5634E-05
	18	<b>2.3071E-07</b>	4.5090E-05	5.8939E-06	3.3440E-05
	21	<b>1.2109E-07</b>	1.3246E-05	7.4884E-06	8.1323E-06
	24	<b>7.5690E-08</b>	1.2843E-05	3.3090E-06	5.4421E-06

show that adaptive mutation rate adjustment in PIBEA allows non-dominated individuals to have more stable convergence toward the Pareto-optimal front.

## V. CONCLUSION

This paper studies an EMOA that leverages a new quality indicator, called the prospect indicator, for parent selection and environmental selection operators. Experimental results show that PIBEA effectively performs its operators in high dimensional problems and outperforms three existing well-known EMOAs, NSGA-II, SPEA2 and AbYSS, in terms of convergence velocity, diversity of individuals, coverage of the Pareto front and performance stability.

## REFERENCES

- [1] R. C. Purshouse and P. J. Fleming, "Evolutionary Many-Objective Optimization: An Exploratory Analysis," in *Proc. of IEEE Congress on Evolutionary Computation*, 2003.
- [2] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evol. Computat.*, vol. 2(3), 1994.
- [3] C. C. Coello, "Evolutionary multi-objective optimization: Some current research trends and topics that remain to be explored," *Front. Computat. Sci. China*, vol. 3, no. 1, 2009.
- [4] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, 2007.
- [5] E. Zitzler and S. Knzli, "Indicator-based selection in multiobjective search," in *Proc. of Int'l Conf. on Parallel Problem Solving from Nature*, 2004.
- [6] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Computat.*, vol. 19, no. 1, 2011.
- [7] E. Zitzler, L. Thiele, and J. Bader, "SPAM: Set preference algorithm for multiobjective optimization," in *Proc. of Int'l Conference on Parallel Problem Solving From Nature*, 2008.
- [8] E. Zitzler, D. Brockhoff, and L. Thiele, "The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration," in *Proc. of Int'l Conference on Evolutionary Multi-Criterion Optimization*, 2007.
- [9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans Evol. Computat.*, vol. 6, no. 2, 2002.
- [10] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization," in *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems*, 2002.
- [11] A. J. Nebro, F. Luna, E. Alba, B. Dorronsoro, J. Durillo, and A. Beham, "AbYSS: adapting scatter search for multiobjective optimization," *IEEE Trans Evol. Computat.*, vol. 12, no. 4, 2008.
- [12] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," in *Proc. of Int'l Conference on Parallel Problem Solving from Nature*, 2000.
- [13] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Trans. Evol. Computat.*, vol. 3, no. 4, 1999.
- [14] M. Srinivas and L. Patnaik, "Adaptive probabilities of crossover and mutation in genetic algorithms," *IEEE Trans Syst. Man Cybern.*, vol. 24, no. 4, 1994.
- [15] A. L. Buczaka and H. Wangb, "Optimization of fitness functions with non-ordered parameters by genetic algorithms," in *Proc. of IEEE Congress on Evolutionary Computation*, 2001.
- [16] D. Thierens and D. Thierens, "Adaptive mutation rate control schemes in genetic algorithms," in *Proc. of IEEE Congress on Evolutionary Computation*, 2002.
- [17] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, "Theory of the hypervolume indicator: Optimal  $\mu$ -distributions and the choice of the reference point," in *Proc. of ACM Int'l Workshop on Foundations of Genetic Algorithms*, 2009.
- [18] K. Deb, K. Sindhya, and T. Okabe, "Self-adaptive simulated binary crossover for real-parameter optimization," in *Proc. of ACM Int'l Genetic and Evolutionary Computation Conference*, 2007.
- [19] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization*. Springer, 2005, ch. 6.
- [20] J. Knowles and D. Corne, "The pareto archived evolution strategy: a new baseline algorithm for pareto multiobjective optimisation," in *Proc. of IEEE Congress on Evolutionary Computation*, 1999.
- [21] F. Glover, "A template for scatter search and path relinking," in *Proc. of European Conference on Artificial Evolution*, 1998.
- [22] J. Durillo, A. Nebro, and E. Alba, "The jMetal framework for multi-objective optimization: Design and architecture," in *Proc. of IEEE Congress on Evolutionary Computation*, 2010.
- [23] D. V. Veldhuizen and G. Lamont, "Multiobjective evolutionary algorithm research: A history and analysis," Dept. Elec. Computat. Eng., Air Force Inst. Technol., Tech. Rep. TR-98-03, 1998.