2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

A Multiple Reference Point-based Evolutionary Algorithm for Dynamic Multi-objective Optimization with Undetectable Changes

Radhia Azzouz, Slim Bechikh, and Lamjed Ben Said
SOIE Lab, University of Tunis, Tunisia.
{azouzradhia, slim.bechikh}@gmail.com, lamjed.bensaid@isg.rnu.tn

***Abstract*—Dynamic multi-objective optimization problems involve the simultaneous optimization of several competing objectives where the objective functions and/or constraints may change over time. Evolutionary algorithms have been considered as popular approaches to solve such problems. Despite the considerable number of studies reported in evolutionary optimization in dynamic environments, most of them are restricted to the single objective case. Moreover, the majority of dynamic multi-objective optimization algorithms are based on the use of some techniques to detect or predict changes which is sometimes difficult or even impossible. In this work, we address the problem of dynamic multi-objective optimization with undetectable changes. To achieve this task, we propose a new algorithm called Multiple Reference Point-based Multi-Objective Evolutionary Algorithm (MRP-MOEA) which does not need to detect changes. Our algorithm uses a new reference point-based dominance relation ensuring the guidance of the search towards the Pareto optimal front. The performance of our proposed method is assessed using various benchmark problems. Furthermore, the comparative experiments show that MRP-MOEA outperforms serveral dynamic multi-objective optimization algorithms not only in tracking the Pareto front but also in maintainig diversity over time albeit the changes are undetectable.**

I. Introduction

When dealing with Dynamic Multi-objective Optimization Problems (DMOPs), the optimization algorithm must be able not only to evolve a near-optimal and diverse Pareto Front (PF), but also to continually track time-changing environment. Such finality implies a conflicting requirement of convergence and diversity. Applying Evolutionary Algorithms (EAs) to solve Dynamic Optimization Problems (DOPs) has obtained great attention among many researchers. To the best of our knowledge, the earliest application of EAs to dynamic environments dates back to 1966 [14]. However, it was not until the late 1980s that the subject becomes a research topic. Although many other optimization techniques have been adapted to dynamic environments such as particle swarm optimization [10], [18], artificial immune systems [28], [13], the EA area is still the largest one. EAs were first applied to Dynamic Single-objective Optimization Problems (DSOPs). In fact, convergence during the run may cause a lack of diversity and reduce the adaptability of the algorithm. Thereby, several mechanisms have been proposed to keep diversity in the population. Diversity can be maintained throughout the run [9], [24], or increased after a change detection by taking explicit actions such as reinitialization [11] or hypermutation [22]. Also, many other approaches have been proposed such as memory-based techniques [5], [25], multipopulation approaches [3], [6], predictive methods [20], etc. The main difficulty in the multi-objective case is that the PF of a DMOP may change when the environment changes which makes the task of optimization more difficult. The multi-objective EA should be capable of attaining a fast convergence which also implies a rapid loss of diversity during the optimization process [1]. In contrast to the single-objective case, there are few works dealing with DMOPs which include change prediction techniques [8], [27], memory-based approaches [2], and parallel approaches [15]. The majority of these works are based on some techniques to either detect or predict changes. Nevertheless, sometimes it is difficult or even impossible for the algorithms to detect changes. An example of this is the case where there are only some random sub-areas in the whole search space that change. In such a situation, we are not always able to detect the changes or predict them since we do not know when and where they occur in the search space. Another example can be revealed in dynamic scheduling problems where a constraint change extends the feasible areas in the search space. Such change can not be easily detected since population members remains feasible. Therefore, it is important to develop algorithms that do not need to detect changes or are able to be ready for changes in any time during the search process.

Multiple targeted search-based approaches are based on the use of a set of predefined search directions covering the entire optimal PF. Then, multiple searches can be done in each direction instead of searching the whole search space [21]. Multiple Reference Points (RPs) defined a priori can be also used to implement such approach and to provide an approximation of the entire PF [12]. Using RPs in such a way as a means to improve comparability between solutions is a recent search direction, where there are few works dealing only with multi-objective and many-objective optimization problems. To the best of our knowledge, using this approach to solve DMOPs is not yet explored.

This paper proposes a Multiple Reference Point-based Multi-Objective Evolutionary Algorithm (MRP-MOEA), that uses the framework of NSGA-II procedure [11], to deal with dynamic problems in undetectable environments. In fact, dynamic problems require a fast convergence to track the PF

978-1-4799-1488-3/14/$31.00 ©2014 IEEE

before the next change appears. Thus, by using multiple RPs we can potentially gain comparability between incomparable solutions which may lead to the acceleration of convergence. Besides, our approach is based on a new dominance relation, called Multiple Reference Point-based dominance relation (MRP-dominance), allowing to maintain the compromise between convergence and diversity since a strong diversity is also important to preserve the adaptive exploring abilities of the algorithm. Moreover, an archive population is maintained to conserve best solutions discovered during the search process. To ensure a good distribution of solutions along the PF, a new archive update strategy is proposed. The performance of MRP-MOEA is examined among different benchmark problems characterized by various difficulties. Several comparative experiments are conducted to validate the effectiveness of MRP-MOEA against several recently proposed dynamic multi-objective optimization algorithms. The remainder of this paper is organized as follows. Section 2 describes the proposed algorithm and details its components. Section 3 presents the experimental study. Finally, section 4 concludes the paper and gives some avenues for future research.

II. Proposed Approach

A. Basic framework

The main diffculty in DMOPs, is that the problem changes over time. The optimization algorithm must be able to track and to converge to the PF as soon as possible before the next change appears. Inspired by [12] and [23], the main idea behind our algorithm is to define multiple targeted search (also known as goals) and to seek simultaneously the location of the optimal solutions along these different directions, rathar than searching in the whole search space. Since several optimal points can be found relatively to different RPs generated in a structured manner (discussed later) and covering the entire search space, the algorithm may be able to converge rapidly to the desired PFs. The framework of the proposed algorithm is based on the original NSGA-II algorithm with significant changes in the non-domination sort mechanism and some other extensions such as the use of a local search technique at the beginning of each generation. This is to ameliorate existing solutions and to detect the new search directions whenever a change appears. MRP-MOEA is described in detail in algorithm 1.

Algorithm 1 MRP-MOEA

INPUT: N (population size), G_{max} (maximum number of generations)
OUTPUT: A (archive population)
BEGIN
  Randomly initialize a population POP;
  Evaluate the fitness values of individuals in POP;
  $A \leftarrow$ NonDominated_Set (POP) ;
  $gen \leftarrow 0$;
  $RPs \leftarrow$ Generate_RPs ();
  $t \leftarrow 0$;
  While ($gen < G_{max}$)
    $t \leftarrow$ Update_Time (gen);
    $POP \leftarrow$ Re-evaluate_Objective_Values (POP, t);
    $ToLS \leftarrow$ Select_solutions_toLS (POP); /* select solutions to participate in the local search*/
    $LS \leftarrow$ LocalSearch ($ToLS$);
    $POP \leftarrow$ Merge ($POP \setminus \{ToLS\}$, LS);
    $C \leftarrow$ Crossover (POP);
    $M \leftarrow$ Mutation (C);
    $M \leftarrow$ Evaluate_Objective_Values (M, t) ;
    $POP \leftarrow$ Merge (POP, M); /*combine parent and child populations*/
    $POP \leftarrow$ Normalization (POP);
    $POP \leftarrow$ Assignement_To_RPs (POP, RPs);
    $POP \leftarrow$ Selection (POP); /*using MRP-dominance relation*/
    $A \leftarrow$ Update_Archive (POP);
    $gen \leftarrow gen + 1$;
  End While
  Return A;
END

B. RP set generation

As already mentionned, our algorithm is based on the use of a set of RPs generated in a structured way. It is worth noting that RPs are used in this paper not to present user preferences and to guide the search towards his preference region but to predefine several search directions covering the entire search space in order to accelerate the convergence speed. Thereby, any predefined method that provides widely distributed solutions in the normalized hyperplane can be used. In this work, we adopt Das and Dennis's approach [7]. It generates K points on a normalized hyperplane with a uniform spacing δ in each axis, for any number of objectives M. Let p be the number of divisions considered along each objective (i.e., $p = 1/\delta$), the number of generated solutions K is calculated as follows:

$$K = \binom{M+p-1}{p} \tag{1}$$

Figure 1 shows the set of the generated RPs in a normalized hyperplane for a three-objective problem ($M = 3$) with $\delta = 0.2$ (i.e., $p = 5$). The number of created RPs is equal to 21. The population size of the algorithm is set to the number of RPs. Each solution is affected to its closest RP based on the Euclidean distance calculated using the normalized objective function values. The normalization of population members will be detailed in the following section.

C. Normalization and Assignement of population members to RPs

Let $\overline{z} = (f_1^{min}, f_2^{min}, ..., f_M^{min})$ and $z^e = (f_1^{max}, f_2^{max}, ..., f_M^{max})$ denote respectively, the ideal point and the extreme point in the population. As noted in [12], the objective function values of the population are normalized as follows:

$$f'_j(x) = \frac{f_j(x) - f_j^{min}}{a_j - f_j^{min}}, \ \forall\ j = 1, 2, ..., M \tag{2}$$

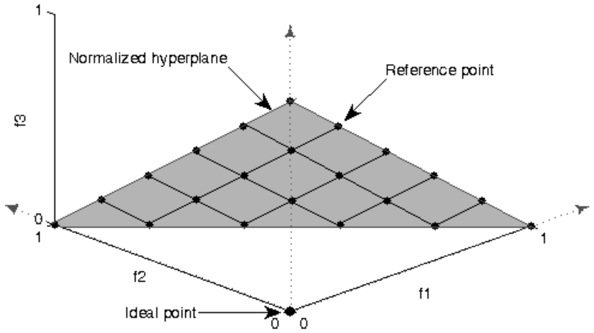

Figure 1. Representation of a set of RPs in a normalized hyperplane with $M = 3$ and $p = 5$

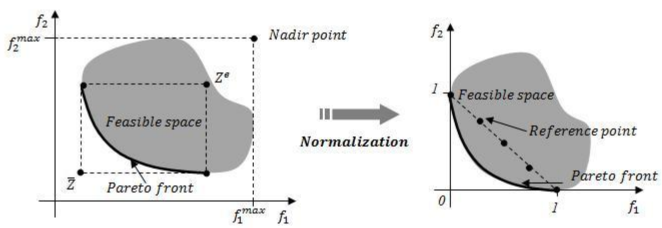

Figure 2. Illustration of a bi-objective space before and after normalization.

$$a_j = 1/\alpha_j \tag{3}$$

$$\alpha = (Z)^{-1}u \tag{4}$$

where u is a M-dimensional vector of ones. The j-th row of the matrix Z is the solution having f_j^{max}. Thereby, a hyperplane is created using the solutions that have led to the coordinates of the extreme point z^e. If any of the a_j's are negative, a_j's are set to f_j^{max}. Figure 2 shows the search space of a bi-objective minimization problem before and after performing normalization with the illustration of generated RPs ($p = 4$) using the strategy described in section II.B.

Each population member is associated with its closest RP. This can be judged using the minimal Euclidean distance calculated between the solution in the normalized space and each RP. The pseudocode of solutions associations with RPs is presented in Algorithm 2.

Algorithm 2 Association procedure

1. **INPUT:** POP (population), RPs
2. **OUTPUT:** POP (assigned population)
3. **BEGIN**
4. For each solution $x \in POP$;
5. For each RP $r \in RPs$;
6. calculate $d\,(x,\, r)$;
7. End For;
8. $ref \leftarrow argmin_{r \in RPs} d\,(x,\, r)$;
9. $Assign\,(x,\, ref)$;
10. End For;
12. Return POP;
13. **END**

D. Multiple Reference Point-based dominance relation

As noted in [26], Pareto-domination can handle both population convergence and population diversity. On the one hand, comparability between solutions accelerates population convergence. On the other hand, indifference between them helps maintaining population diversity. In this work, we propose a new dominance relation that permits to balance between convergence and diversity called Multiple Reference Point-based dominance relation denoted as (MRP-dominance). Let x and y be two solutions in the objective space, x $MRP - dominates$ y, denoted as $x \prec_{MRP-dominance}$ y if and only if:

1) x Pareto domiantes y; or
2) x and y are Pareto equivalent and only x is situated in the neighborhood of at least one RP; or
3) x and y are Pareto equivalent and x and y are not situated in any neighborhood and the number of solutions already chosen for the RP, to which is affected x, is lower than the one for the RP to which is affected y.

Let X be a set of solutions, a solution $x \in X$ is said to be in the neighborhood (means assigned to RP) of a RP r if and only if

$$\frac{d(x,\, r) - d_{min}}{d_{max} - d_{min}} \leq \gamma \tag{5}$$

$$d_{min} = min(d(y,\, r)),\ \forall y \in X \tag{6}$$

$$d_{max} = max(d(y,\, r)),\ \forall y \in X \tag{7}$$

where $d(x,\, r)$ is the Euclidean distance between the solution x and the RP r and γ is a specified threshold.

On the one hand, using MRP-dominance, we note that we raise the comparability level between solutions. On the other hand, the third case may ensure the exploration of the whole search space and to avoid focusing on a particular region. It gives chance to RPs not yet present in the selected population to be represented and to provide, if they are prominent, some solutions that can survive in the following generations. Moreover, MRP-dominance is Pareto compliant, it preserves the order induced by the Pareto dominance relation.

E. Local search

The local search technique accelerates the convergence speed since it improves the individuals of a population to the nearest local optima. We propose to adopt a well directed local search using a binary indicator-based selection method. This may permit to avoid some of the drawbacks of classical methods such as Pareto dominance and aggregation methods. The epsilon indicator $I_\epsilon(x_1,\, x_2)$ represents the minimal translation in the objective space with which x_1 dominates x_2. The $I_\epsilon(x_1,\, x_2)$ is calculated as follows:

$$I_\epsilon(x_1,\, x_2) = max_{j \in \{1,\, \ldots,\, M\}}(f_j(x_1) - f_j(x_2)) \tag{8}$$

Algorithm 3 $\epsilon-$Local Search algorithm

1. **INPUT:** x (current solution), $Q(x)$ (Quality of x),
2. P (initial population), N_{trial}(number of trials),
3. **OUTPUT:** x' (output solution)
4. **BEGIN**
5. $cpt \leftarrow 0$;
6. $stop \leftarrow false$;
7. $x' \leftarrow x$;
8. While (($\neg stop$) and ($cpt <= N_{trial}$)) do
9. $x^* \leftarrow Mutation(x)$;
10. $Q(x^*) \leftarrow I_\epsilon(P\backslash\{x\}, x^*)$;
11. If $((Q(x^*) > Q(x))$
12. $P \leftarrow Insert(P\backslash\{x\}, x^*)$;
13. $P \leftarrow$ Re-evaluate_Quality(P);
14. $stop \leftarrow true$;
15. $x' \leftarrow x^*$;
16. Else
17. $cpt \leftarrow cpt + 1$;
18. End If
19. End While
20. Return x';
21. **END**

The quality of solutions according to a whole population P and a binary indicator I can be evaluated using several different approaches [16]. Since the quality of a solution is defined essentially by the presence of similar or better solutions in the population, we choose to use the following equation in the evaluation of solutions as the acceptance strategy:

$$I(P\backslash\{x\}, x) = min_{z\in P\backslash\{x\}}(I(z, x)) \tag{9}$$

In each generation, $\zeta\%$ of the solutions are considered as the input population P for the local search algorithm. On each solution in P, a local search step ($\epsilon-$local search) is applied. The $\epsilon-$local search algorithm is described in algorithm 3 where a neighbor is generated using a polynomial mutation with a low spread (c.f. line 9). It is worth noting that we use the first improvement strategy. Thereby, the local search step is stopped if the neighbor's indicator value is better than the current solution's indicator value (c.f. line 11). However, if the current solution is better than his N_{trial} neighbors generated successively, the solution is considered as a local optimal solution and it is maintained. The entire local search is terminated when all solutions are either maintained or ameliorated.

F. Archive update

In order to provide well-distributed solutions along the PF, we propose an archive update strategy that permits to maintain representatives of all prominent RPs. As a first step, we select non-dominated solutions among combined population of old archive and the actual population. If the size of the selected population is greater than the archive $maxsize$, we empty the archive and we follow the steps below:

1) **Step 1** : assemble solutions by RPs, the formed subsets are denoted by S_k, where $k = 1, ..., K$ and K is the number of RPs represented in the selected popuation.
2) **Step 2** : calculate $nbsol = max(min\ (\mid S_k \mid), 0)$ if $((nbsol * nbSets) > (maxsize- \mid archive \mid))$ then $nbsol = ceil((maxsize- \mid archive \mid)/nbsets)$ where $nbSets$ is the number of not empty S_k.
3) **Step 3** : select $nbsol$ solutions from all S_k and add selected solutions to the archive.
4) **Step 4** : remove selected solutions from all S_k.
5) **Step 5** : if $\mid archive \mid < maxsize$, return to step 2.

Figure 3 illustrates an example of archive update with five RPs r_i; $i = 1, 2, ..., 5$ where the number of solutions associated to the RP r_5, is equal to zero.

III. EXPERIMENTAL STUDY

A. Performance metrics

1) Convergence indicators:

- The variable space generational distance metric (VD)
  The VD metric measures the closeness of the approximated PF to the optimal PF [2].
- The Inverted Generational Distance metric (IGD)
  In addition to the gap between the optimal PF and the optimized one, the IGD metric [19] can measure the diversity of the obtained PF.
- The HyperVolume (HV)
  The hypervolume of a set A with respect to a RP ref denoted by $HV(A, ref)$ is the hyperarea of the set $R(A, ref)$. $HV(A, ref)$ measures how much of the objective space is dominated by A [4].

Let A be the obtained PF and P^* a set of uniformly distributed points along the optimal PF in the objective space, the GD, the IGD and the hypervolume ratio ($HVRatio$) metrics can be defined as follows

$$GD(A, P^*) = \frac{\sqrt{\mid A \mid \sum_{v\in A} d(v, P^*)^2}}{\mid A \mid} \tag{10}$$

$$IGD(A, P^*) = \frac{\sum\limits_{w\in P^*} d(w, A)}{|P^*|} \tag{11}$$

$$HVRatio(A, ref) = \frac{HV(A, ref)}{HV(P^*, ref)} \tag{12}$$

where $d(v, P^*)$ and $d(w, A)$ are respectively the minimum Euclidean distance between v and the points in P^*, and between w and the points in A.

2) Diversity indicator:

- The maximum spread metric (MS)
  The adaptation of the maximum spread metric to dynamic multi-objective optimization (MS') was introduced in [2] and is defined as follows

$$MS'(A, P^*) = \sqrt{\frac{1}{M}\sum_{j=1}^{M}(\frac{min(A_{j,u}, {P^*}_{j,u}) - max(A_{j,l}, {P^*}_{j,l})}{{P^*}_{j,u} - {P^*}_{j,l}})^2} \tag{13}$$

where $A_{j,u}$ and $A_{j,l}$ are respectively the maximum and the

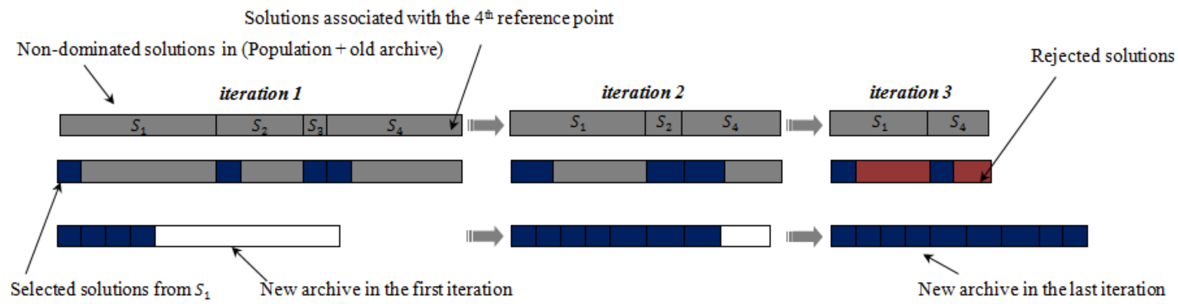

Figure 3. Example of archive update with five reference points.

Table I
PARAMETER SETTING FOR DIFFERENT ALGORITHMS.

Parameters	**Settings**
Populations	Population size = 100 Archive size = 100
Selection	Binary Tournament selection + crowed-comparison-operator
Crossover Operator	Simulated BinaryCrossover with a distribution index of 20
Mutation Operator (Dy-NSGA-II)	Polynomial mutation with a distribution index of 20
Mutation Operator (LS-Strategy)	Polynomial mutation with a distribution index of 100
Ratio ζ	10%
Threshold γ	0.25
Termination criterion	20,000 evaluations for comparative experiments 1000 generations for the others

minimum value of the j-th objective in the obtained PF. $P^*_{j,u}$ and $P^*_{j,l}$ are respectively the maximum and the minimum value of the j-th objective in the optimal PF. MS' is applied to measure how well the optimal PF is covered by the obtained PF. A higher value of MS' reflects that a larger area of P^* is covered by A.

In this work, for all performance indicators we use the mean indicator over time. Let θ be an indicator, the mean θ, denoted by $\overline{\theta}$, is calculated as follows

$$\overline{\theta} = \frac{1}{nbChanges} \sum_{i=1}^{nbChanges} \theta_i \tag{14}$$

where $nbChanges$ is the number of occured changes and θ_i is the θ value calculated before the occurence of the $(i+1)th$ change.

B. Experimental results

In this section, simulation results are presented using several DMOPs to assess the performance of the algorithm facing different difficulties. Parameter settings are presented in Table I. For each experiment, thirty one independent simulation runs with randomly generated initial populations are conducted. The experiments are performed at different severity levels of $\{1, 10\}$ and different frequencies of $\{5, 10, 25, 50\}$ to study the performance of our algorithm in different challenging environments.

1) Effect of increasing change severity and change frequency:

a) FDA1: This set of experiments is performed in order to demonstrate the effect of increasing the change severity and the change frequency on the proposed algorithm for type I DMOPs. The benchmark problem used is FDA1 test problem where the number of decision variables is set to 10 [17]. In FDA1 test problem, which is a type I DMOP, the optimal Pareto set changes over time while the optimal PF remains invariant. This problem has a convex optimal PF.

b) dMOP1 and dMOP2: This set of experiments is performed in order to evaluate the performance of the algorithm facing a type II DMOP and a type III DMOP in various change severity and change frequency levels. The benchmark problems used are dMOP1 and dMOP2 test problems [2]. The test function of dMOP1 is a Type III DMOP where only the PF is dynamic which changes from convex to concave. The test function of dMOP2 is a Type II problem where both the Pareto set and the PF change over time. Table II presents the obtained results for the $\overline{IGD}$, the $\overline{HVRatio}$ and the $\overline{MS'}$ metrics for respectively FDA1, dMOP1 and dMOP2 test problems for 1000 generations.

On the one hand, we can notice in Table II that with a low change severity ($n_t = 10$), MRP-MOEA presents better results. This may be explained by the fact that with slight changes, past optimal solutions remain prominent after change appearance. Also, the local search procedure guides the population towards the new search directions and it ameliorates population when an environmental change takes place. Moreover, we notice that change severity does not affect the performance of MRP-MOEA for dMOP2 test problem, instead, they are improved in both aspects of convergence and diversity. This observation may validate the adaptability capacity of MRP-MOEA. When changes are large ($n_t = 1$), it is observed that the quality of results does not decrease dramatically. This observation may be explained by the good level of diversity maintained all over the search progress. This is due to (1) the balance between convergence and diversity ensured by the MRP-dominance relation and (2) the archive update strategy. On the other hand, with the decrease of the change frequency, the time dedicated to adaptation is more important and gives the algorithm the ability to converge to the optimal PF. In fact, as presented in [11], a hypervolume ratio smaller than 94% is considered to be a threshold for indicating a poor performance. Thus, examining the variation

Table II
$\overline{IGD}$, $\overline{HVRatio}$ AND $\overline{MS'}$ METRICS FOR FDA1, DMOP1 AND DMOP2 TEST PROBLEMS OVER 1000 GENERATIONS.

$(\tau_T,\, n_T)$		FDA1 $\overline{IGD}$	FDA1 $\overline{HVRatio}$	FDA1 $\overline{MS'}$	dMOP1 $\overline{IGD}$	dMOP1 $\overline{HVRatio}$	dMOP1 $\overline{MS'}$	dMOP2 $\overline{IGD}$	dMOP2 $\overline{HVRatio}$	dMOP2 $\overline{MS'}$
(5, 10)	Med	0.0504	0.8800	0.9438	0.0113	0.9835	0.9951	0.0347	0.9951	0.9552
	IQR	0.0017	0.0039	0.0030	0.0002	0.0028	2.21E-05	0.0058	0.0007	0.0022
(10, 10)	Med	0.0277	0.9336	0.9735	0.0021	0.9944	0.9998	0.0149	0.9988	0.9840
	IQR	0.0078	0.0182	0.0094	4.57E-19	0.0001	1.74E-05	0.0061	0.0010	0.0102
(25, 10)	Med	0.0117	0.9711	0.9906	0.0020	0.9952	0.9997	0.0070	0.9998	0.9937
	IQR	0.0032	0.0073	0.0032	4.98E-19	0.0001	9.10E-06	0.0020	0.0005	0.0023
(50, 10)	Med	0.0080	0.9807	0.9944	0.0018	0.9953	0.9999	0.0066	0.9999	0.9947
	IQR	0.0005	0.0012	0.0006	5.22E-19	0.0001	8.98E-06	0.0017	3.7E-06	0.0031
(25, 1)	Med	0.0178	0.9568	0.9835	0.0024	0.9936	0.9997	0.0052	0.9999	0.9991
	IQR	0.0005	0.0011	0.0010	4.98E-19	0.0001	4.53E-05	0.0009	0.0001	0.0003

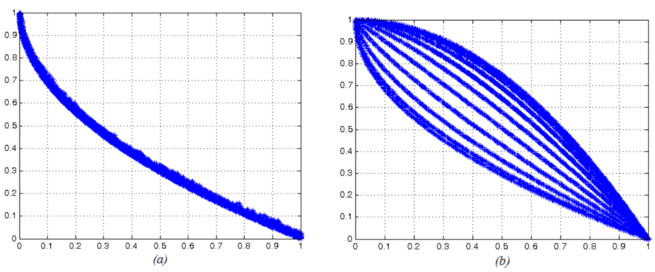

Figure 4. The obtained PFs for (a) FDA1 and (b) dMOP1 over 1000 generations.

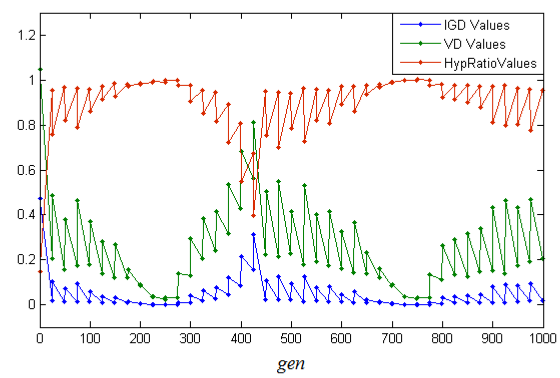

Figure 5. IGD, VD and $HVRatio$ curves for FDA1 over 1000 generations.

of the obtained $\overline{HVRatio}$, we can state that MRP-MOEA is able to handle the increase of the change frequency even for $\tau_t = 5$ except FDA1 problem which does not present a good $\overline{HVRatio}$ with $\tau_t \leq 10$. Concerning the $\overline{MS'}$, variations are slightly significant. Moreover, when observing the variation of the $\overline{IGD}$, it is clearly observed that the obtained results slightly deteriorate mainly for $\tau_t = 5$ which is explained by the decrease of the time devoted to optimization and adaptation.

Figure 4 shows the obtained PFs for FDA1 and dMOP1 test problems for 1000 generations with $\tau_t = 25$ and $n_t = 10$. It can be noticed that the algorithm is able to track and to converge to the PF as it changes over time without the need to detect changes.

2) Study of the adaptability of MRP-MOEA with environment changes: Figure 5 illustrates the changes of IGD, VD and $HVRatio$ during the optimization process for FDA1 test problem over 1000 generations and with $\tau_t = 25$ and $n_t = 10$. As we can notice, the three curves are in total harmony. In fact, any increase of the $HVRatio$ is accompanied by a decrease of IGD and VD while any reduction of $HVRatio$ is accompanied by an increase of the latter ones. Moreover, Figure 6 shows the evolution of VD, IGD and MS' metrics over 200 generations for FDA1, dMOP1 and dMOP2 problems with $\tau_t = 25$ and $n_t = 10$. It is clearly observed that the occurrence of a change $((gen \ mod \ 25) = 0)$ automatically leads to a deterioration in performance. Our algorithm has shown a capacity of adaptability since after a small number of generations and without explicitly detecting changes, it recovers a good level of performance. This ability may be explained by the good level of diversity maintained by the algorithm all over the search progress.

3) Study of the influence of the number of RPs in the performance of MRP-MOEA: Our algorithm is based on the predefinition of a set of RPs to be used as targeted search directions. Thereby, the more the number of RPs is, the more the search space is covered. However, a very large number may slow down convergence. In this section, we aim to study the influence of the size of the RP set in the performance of MRP-MOEA. Thereby, we conduct a set of experiments on dMOP1 and dMOP2 test problems with several settings of the number of RPs. Table III presents the obtained values of $\overline{IGD}$ and $\overline{MS'}$ metrics with $\tau_t = 10$ and $n_t = 10$ over 1000 generations where N is the size of the population.

We observe that when the number of RPs is smaller than the size of the population, the performance of MRP-MOEA decreases in both aspects of convergence and diversity. Moreover, when the number of RPs exceeds N, this does not provide a significant improvement while it may slow down convergence since it increases the incomparability between solutions.

4) Comparative experiments: This subsection is mainly devoted to confront MRP-MOEA to one of the most recent works

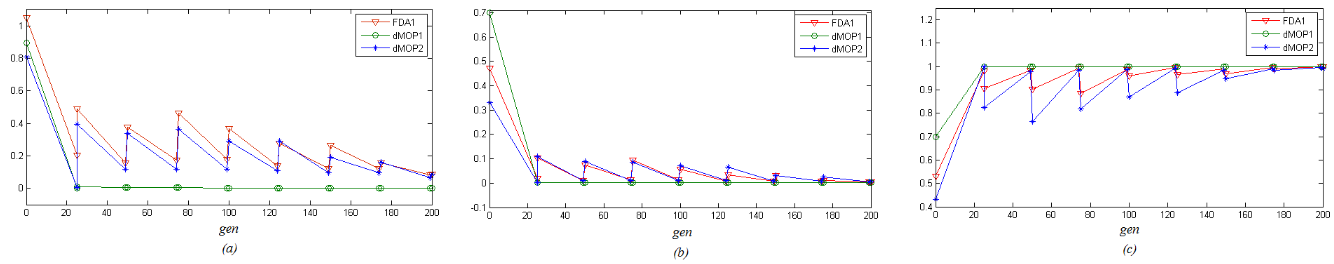

Figure 6. Evolution of (a) VD, (b) IGD and (c) MS' metrics over 200 generations for FDA1, dMOP1 and dMOP2 problems.

Table III
THE INFLUENCE OF THE NUMBER OF RPS IN THE PERFORMANCE OF MRP-MOEA ON DMOP1 AND DMOP2 TEST PROBLEMS.

NB RPs	dMOP1		dMOP2	
	$\overline{IGD}$	$\overline{MS'}$	$\overline{IGD}$	$\overline{MS'}$
$N/2$	0.0050	0.9849	0.0168	0.9840
N	0.0021	0.9998	0.0149	0.9940
$2N$	0.0011	0.9999	0.0142	0.9944
$3N$	0.0007	0.9999	0.0137	0.9947

in this research area, called dynamic competitive-cooperation coevolutionary algorithm (dCOEA) [2]. dCOEA is a coevolutionary multi-objective algorithm based on competitive and cooperative mechanisms. In fact, the problem is decomposed into several subcomponents along the decision variables. These subcomponents are optimized by different species subpopulations through an iterative process of competition and cooperation. The optimization of each subcomponent is no longer restricted to one species but at each cycle, different competing species solve a single component as a collective unit which permits the discovery of interdependencies among the species. In order to deal with dynamic multi-objective optimization environments, authors have proposed to: (1) introduce diversity via stochastic competitors, (2) start the competitive mechanism, whenever a change is detected, independently of its fixed schedule and (3) handle outdated archived solutions using an additional external population in addition to the archive. The main advantages of the proposed algorithm are: (1) it overcomes limitations in conventional coevolutionary models by incorporating both competitive and cooperative mechanisms allowing the problem decomposition to emerge simultaneously with the optimization process and (2) it expoilts the high speed of convergence in coevolution allowing the algorithm to adapt quickly to changing environments. However, the main drawback of this algorithm is the need to explicitly detect changes.

To examine its performance, dCOEA was compared against two different dynamic MOEAs denoted as MOEA and dCCEA [2], respectively. In this study, we compare MRP-MOEA to dCOEA and its competitors (MOEA and dCCEA). For each test problem and each pair of algorithms, we perform a (two-sided) Wilcoxon test to decide whether or not the difference between the indicated median values (for each performance indicator) for the two algorithms is statistically significant on the considered problem instance. The results of the significance tests for the pairwise comparisons at level $\alpha = 0.05$ are presented in the form of (-: no significance) and (+: significance) in the order on which the algorithms appear. Table IV illustrates the obtained results for $\overline{VD}$ and $\overline{MS'}$ metrics for dMOP1 and dMOP2 test problems after $20,000$ evaluations.

Table IV shows that MRP-MOEA outperforms MOEA and dCCEA in both aspects of convergence and diversity for all test problems and upon different frequency and severity levels. Also, we notice that dCCEA is unable to find a diverse PF when the shape of PF is dynamic. When compared to dCOEA, we also observe that MRP-MOEA produces better results for dMOP2 test problem for different settings except the settings of $n_t = 10$ and $\tau_t = 5$ where dCOEA provides a better diversity level. It is important to note that these results are obtained despite of the fact that dCOEA uses a diversity scheme consisting on starting the competitive mechanism, whenever a change is detected which is not the case for MRP-MOEA which does not take any explicit action when a change takes place. When observing results relative to dMOP1, we note that while dCOEA provides smaller $\overline{VD}$ values for $n_t = 10$ and $\tau_t = 5$ and $\tau_t = 25$, we can see that MRP-MOEA maintains the best diversity among the different combinations of severity and frequency values. It is able to find the most diversified PF despite not having the best convergence. Moreover, it provides the most important $\overline{VD}$ values for all other settings.

IV. CONCLUSION

In this paper, we have proposed a new multiple search direction-based dynamic multi-objective EA to seek simultaneously the location of the optimal solutions along different directions, rathar than searching in the whole search space. Our algorithm called MRP-MOEA is able to deal with undetectable changes since it does not need to detect or predict them. MRP-MOEA is based on the use of a new Multiple Reference Point based-dominance relation called MRP-dominance that balances between convergence and diversity. Also, we have proposed a new archive update strategy that provides well distributed solutions along the PF. Moreover, to improve the adaptability of the algorithm, we have opted to the use of a new local search technique that guides the search towards new search directions whenever a change occurs. Experiments have shown the ability of our algorithm to converge and track PFs over time thanks to the good level of diversity

Table IV
COMPARATIVE RESULTS FOR $\overline{VD}$ AND $\overline{MS'}$ METRICS FOR DMOP1 AND DMOP2 TEST PROBLEMS. BEST PERFORMANCE IS SHOWN IN BOLD.

		dMOP1								dMOP2							
		MOEA		dCCEA		dCOEA		MRP-MOEA		MOEA		dCCEA		dCOEA		MRP-MOEA	
$(\tau_T,\ n_T)$		$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$	$\overline{VD}$	$\overline{MS'}$
(5, 10)	Med	0.122	0.917	0.252	0.828	0.0082	0.979	0.0091	0.9945	0.657	0.983	0.230	0.881	0.363	0.989	0.1348	0.9522
	IQR	0.020	0.045	0.032	0.031	0.0035	0.015	0.001	1.71E-05	0.034	0.011	0.017	0.025	0.028	0.007	0.021	0.0032
		(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+-+)	(+++)	(+++)	(+++)	(-++)	(+++)	(+++)
(10, 10)	Med	0.121	0.912	0.132	0.878	0.003	0.99	0.0029	0.9998	0.506	0.984	0.165	0.912	0.173	0.9921	0.0705	0.994
	IQR	0.032	0.017	0.026	0.009	0.0015	0.005	0.0001	4.83E-05	0.004	0.109	0.026	0.034	0.017	0.0051	0.004	0.0082
		(-++)	(+++)	(-++)	(+++)	(++-)	(+++)	(++-)	(+++)	(+++)	(+++)	(+-+)	(+++)	(+-+)	(+++)	(+++)	(+++)
(25, 10)	Med	0.067	0.944	0.032	0.943	0.0015	0.9916	0.0021	0.9996	0.604	0.985	0.069	0.958	0.061	0.9947	0.0503	0.9948
	IQR	0.023	0.045	0.009	0.029	0.0004	0.0073	0.0002	4.53E-05	0.095	0.0112	0.021	0.015	0.012	0.0048	0.0047	0.0031
		(+++)	(-++)	(+++)	(-++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+-+)	(+++)	(+-+)	(++-)	(+++)	(++-)
(10, 1)	Med	0.116	0.923	0.1310	0.890	0.0032	0.989	0.003	0.9939	1.154	0.982	0.192	0.895	0.143	0.9971	0.058	0.9991
	IQR	0.025	0.047	0.0185	0.032	0.0023	0.0064	0.0002	1.5E-05	0.042	0.022	0.026	0.032	0.035	0.0076	0.0147	0.0003
		(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)	(+++)

maintained all over the search progress. When confronted to serveral dynamic multi-objective optimization algorithms, MRP-MOEA has shown competitive and better results in terms of convergence and diversity. However, it should be noted that in this paper, we are only interested to unconstrained problems. Hence, a first future research direction is to adapt MRP-MOEA to deal with problems with dynamic constraints. Moreover, we plan to apply MRP-MOEA to real world problems such as dynamic multi-objective scheduling and transportation problems.

REFERENCES

[1] A. A. Tantar, E. Tantar, P. Bouvry, "A classification of dynamic multi-objective optimization problems," Genetic and Evolutionary Computation Conference (Companion), pp. 105-106, 2011.

[2] C. K. Goh and K. C. Tan, "A competitive-cooperative coevolutionary paradigm for dynamic multi-objective optimization," IEEE Transcations on Evolutionary computation, vol. 13, no. 1, pp. 103-127, 2009.

[3] C. Li and S. Yang, "A general framework of multipopulation methods with clustering in undetectable dynamic environments," IEEE Transactions on Evolutionary Computation, 2011.

[4] D.A. van Veldhuizen, "multi-objective evolutionary algorithms: classification, analyses, and new innovations," PhD thesis, Graduate School of engineering Air University, 1999.

[5] D. E. Goldberg and R. E. Smith, "Non stationary function optimization using genetic algorithms with dominance and diploidy," Genetic Algorithms, J. J. Grefenstette, Ed: Lawrence Erlbaum, pp. 59-68, 1987.

[6] F. Oppacher and M. Wineberg, "The shifting balance genetic algorithm: Improving the ga in a dynamic environment," Genetic and Evolutionary Computation Conference, vol.1, pp. 504-510, 1999.

[7] I. Das, J.E. Dennis, "Normal-bounday intersection: A new method for generating Pareto optimal points in multicriteria optimization problems," SIAM Journal on Optimization, vol.8, pp. 631-657, 1998.

[8] I. Hatzakis and D. Wallace, "Dynamic multi-objective optimization with evolutionary algorithms: A foward-looking approach," Genetic Evolutionary Computation Congress, pp. 1201-1208, 2006.

[9] J. J. Grefenstette, "Genetic algorithms for changing environments," Parallel Problem Solving from Nature, R. Maenner and B. Manderick, Eds. Amsterdam, The Netherlands: North-Holland, pp.137-144, 1992.

[10] J. Wei, L. Jia, "A novel particle swarm optimization algorithm with local search for dynamic constrained multi-objective optimization problems," IEEE Congress on Evolutionary Computation pp. 2436-2443, 2013.

[11] K. Deb, U.B.N. Rao, S. Karthik, "Dynamic multi-objective optimization and decision-making using modified NSGA-II: a case study on hydrothermal power scheduling," Evolutionary Multi-Criterion Optimization, LNCS, vol. 4403, pp. 803-817, 2007.

[12] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving problems with box constraints," IEEE Transactions On Evolutionary Computaion, 2013.

[13] K. Trojanowski and S. T. Wierzchon, "Immune-based algorithms for dynamic optimization," Information Sciences, vol. 179, no. 10, pp. 1495-1515, 2009.

[14] L. J. Fogel, A. J. Owens, and M. J. Walsh, "Artificial intelligence through simulated evolution," New York: Wiley, 1966.

[15] M. Cámara, J. Ortega and F. de Toro, "Parallel processing for multi-objective optimization in dynamic environments," IEEE International Parallel and Distributed Processing Symposium, pp. 1-8, 2007.

[16] M. Basseur, A. Liefooghe, K. Le, E.K. Burke, "The efficiency of indicator-based local search for multi-objective combinatorial optimization problems," Journal of Heuristics, Vol. 18, no. 2, pp. 263-296, 2012.

[17] M. Farina, P. Amato, and K. Deb, "Dynamic multi-objective optimization problems: Test cases, approximations and applications," IEEE Transactions on Evolutionary Computation, vol. 8, no. 5, pp. 425-442, 2004.

[18] M. Helbig, Andries Petrus Engelbrecht, "Dynamic multi-objective optimization using PSO," Metaheuristics for Dynamic Optimization, pp. 147-188, 2013.

[19] M. Sierra, C. Coello Coello, "Improving pso-based multi-objective optimization using crowding, mutation and epsilon-dominance," Third International Conference on Evolutionary multi-criterion optimization, vol. 3410 of Lecture Notes in Computer Science, pp. 505-519, 2005.

[20] P.A.N. Bosman, "Learning and anticipation in online dynamic optimization," Evolutionary Computation in Dynamic and Uncertain Environments, Vol. 51, pp. 129-152, 2007.

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

[22] R. W. Morrison, and K. A. De Jon, "Triggered hypermutation revisited," Congress on Evolutionary Computation, vol. 2, pp. 1025-1032, 2000.

[23] S. Bechikh, "Incorporating decision maker's preference information in evolutionary multi-objective optimization," PhD thesis, University of Tunis, Tunisia, 2013.

[24] S. Yang, "Genetic algorithms with memory and elitism-based immigrants in dynamic environments," Evolutionary Computation, vol. 16, no. 3, pp. 385-416, 2008.

[25] S. Yang and X. Yao, "Population-based incremental learning with associative memory for dynamic environments," IEEE Transactions on Evolutionary Computation, vol. 12, no. 5, pp. 542-561, 2008.

[26] S. Zeng, D. Zhou and H. Li, "Non-dominated sorting genetic algorithm with decomposition to solve constrained optimization problems," International Journal of Bio-Inspired Computation, Vol. 5, No. 3, 2013.

[27] W.T. Koo, C.K. Goh, and K.C. Tan, "A predictive gradient strategy for multi-objective evolutionary algorithms in a fast changing environment," Memetic Computing, pp. 87-110, 2010.

[28] Z. Zhang "Multi-objective optimization immune algorithm in dynamic environments and its application to greenhouse control," Applied Soft Computing, vol. 8, no. 2, pp. 959-971, 2008.