A Decomposition-Based Interactive Method for Multi-Objective Evolutionary Algorithm

Long NGUYEN and Lam Thu BUI
Faculty of Information Technology, Le Quy Don Technical University, Ha noi, Vietnam
Emails: longit76@mta.edu.vn, lam.bui07@mta.edu.vn

***Abstract*: Multi-objectivity has existed in many real-world optimization problems. In most multi-objective cases, objectives are often conflicting, there is no single solution being optimal with regards to all objectives. These problems are called *Multi-objective Optimization Problems (MOPs)*. To date, there have been a large number of methods for solving MOPs including evolutionary methods (namely *Multi-Objective Evolutionary Algorithms MOEAs*). With the use of a population of solutions for searching, MOEAs are naturally suitable for approximating optimal solutions (called the *Pareto Optimal Set (POS)* or the *efficient set*). There has been a popular trend in MOEAs considering the role of Decision Makers (DMs) during the optimization process (known as the *human-in-loop*) for checking, analyzing the results and giving the preference to guide the optimization process. This is called the *interactive method*.**

In this paper, we propose an interactive approach integrating with a generic framework of MOEA/D, a widely-used and decomposition-based MOEA. Basically, MOEA/D decomposes a multi-objective optimization problem into a number of different single-objective sub-problems and defines neighborhood relations among these sub-problems. It was a population-based method to optimize the sub-problems simultaneously. Each sub-problem is optimized by using information mainly from its neighboring sub-problems. In MOEA/D, an ideal point is used to choose neighborhood solutions for each run.

In our proposal, we change the reference point, which originally is defined as the best (found by MOEA/D) towards the point being given by DM. The reference point from DM is used to either replace or adjust the current ideal point obtained by MOEA/D. We carry out a case study on several test problems and obtained quite good results.

Keywords: MOEA, interactive, reference point, MOEA/D.

I. INTRODUCTION

When solving MOPs, we need to simultaneously optimize several objective functions [2]. As a result, we usually obtain several *trade-off solutions*, which are called *Pareto optimal solutions*. Methods for multi-objective optimization can be classified into several classes including the *Interactive method*. With the Interactive method, DM iteratively directs the searching procedure by indicating his/her preference information over the set of solutions until DM satisfies or prefers to stop the process [9]. This method has drawn a significant number of considerations from the research community since it can allow focusing on a particular area of Pareto set, and during the solution process, DM is able to learn about the underlying problem as well as his/her own preference.

To date, many interactive methods have been proposed for solving MOPs [3, 4, 5, 6, 7, 8, 10]. It is worth to note that the aim of the interactive method is to find most suitable solution with regards of DM's preference. An interactive method requires a mechanism to support DM in formulating her/his preferences and identifying preferred solutions in the set of Pareto optimal solutions.

MOEAs can offer a population of solutions at the end. When incorporating the preference information, it can change the fitness function and entirely the selection process as a result. Hence, for each MOEA, there is usually a unique mechanism for interaction being employed. In this paper, we introduce an interactive method integrating with MOEA/D [11], a decomposition-based MOEA. With our proposal, the ideal point, (also called *the reference point*) requested and calculated by MOEA/D, will be modified according to a point being given by DM instead of using the best point obtained by MOEAD/D itself. After DM has specified a reference point, Pareto optimal solutions are found that best correspond to it. If DM is not satisfied, he/she can specify another reference point.

Our main contribution is that we develop two approaches for adjusting the reference point from DM within the framework of MOEA/D. The first approach is to replace the ideal point by the reference point given by DM. For MOEA/D, the ideal point is approximated by finding the best value for each objective. Here, we assume that the ideal point will be set depending on DM. The second one combines the ideal point and the reference point. This method is a compromise between DM's objectivity and the best found by MOEA/D.

In the remainder of the paper, Section II briefly describes multi-objective optimization interactive methods using reference point. Overview of interactive MOEAs is given in Section III. In Section IV, we have a description for MOEA/D, while Section V is dedicated to our methodology for an interactive with MOEA/D. In Section VI we present simulation results and discussions. Finally, the conclusion of this paper is drawn in Section VII.

II. Interactive Methods for Multi-Objective Optimization

Differences between interactive methods are (1) the type of information being given to DM, (2) the way to present information to DM, and (3) transformation of objectives with regards to DM's preference. Among various methods, the preference-point one is the most popular and influential. A typical example is the reference-point interactive method suggested by Wierzbicki [1]. The reference point is to show the aspiration levels of DMs and it can be feasible or infeasible in the objective space. The idea is to transform the multi-objective problem into a series of single objective problems defined by achievement scalarizing functions (derived by reference points from DMs). Mathematically, a reference point z^* is given for an *m-objective* optimization problem of

$$minimizing\ (f_1(x), \ldots, f_m(x)) \quad (1)$$

subject to $x \in S$

with x is the decision variable vector, f_i is the objective function i, S is the feasible search space, and m is the number of objectives.

Then a single-objective optimization problem is defined as the following:

$$minimize\ max_{i=1}^{m}[w_i(f_i(x) - z_i^*)] \quad (2)$$

subject to $x \in S$, with w is the weight vector

Assuming DM is given some information about the problem, the algorithm for this method is described in five steps (see [1] for more details):

Step 1: Present information to DM. Set $h=1$ (h is counter variable)

Step 2: Ask DM to specify a reference point z_*^h

Step 3: Minimize achievement function (see Eq. 2) and obtain a Pareto optimal solution x^h and a corresponding point z^h in the objective space. Present z^h to DM

Step 4: Calculate k other Pareto optimal solutions by perturbing the reference point. Note that new reference points are perturbed as follows:

$\bar{z}(i) = z_*^h + d^h e^i$, where $d^h = \|z_*^h - z^h\|$ and e^i is the i^{th} unit vector.

Step 5: Present the alternatives to DM. If DM found the final solution, then stop the process. Otherwise, ask DM to specify new z_*^{h+1}. Set $h = h+1$ and go **Step 3**.

It is worth to note that, by the way of using the series of reference points (perturbing the current reference points), DM has a chance to evaluate and understand the region of Pareto Optimality. That is why interactive methods are attractive in reality.

III. Interactive MOEAs

In this section, we summarize several typical works on the area of MOEAs with interaction. In [5], the authors proposed an interactive MOEA using a concept of the reference point with their own proposed MOEA (NSGA-II) and finding a set of preferred Pareto optimal solutions near the regions of interest to a DM. In particular, they suggested two approaches: The first is to modify NSGA-II for effectively solving 10-objective problems. The other is to use the hybrid-EMO methodology in allowing the DM to solve multi-objective optimization problems better and with more confidence.

In other works, the *reference direction* is used to guide the search process [3]. The main contribution of this work is that it proposes to supply more than one reference directions in the objective space. The proposed method is exploited to find a set of efficient solutions corresponding to a number of reference directions. This procedure is continued till no further improvement is possible. It is demonstrated on a set of test problems having from two to ten objectives and on an engineering design problem. Results are verified with theoretically exact solutions on two-objective test problems.

An interactive version of the decomposition based multi-objective evolutionary algorithm (iMOEA/D) is also discussed in [6]. For iMOEA/D, an MOP in question is converted into a number of scalar optimization problems by the Tchebycheff decomposition approach with even spread weight vectors. At each interaction, iMOEA/D offers a set of current solutions and asks the DM to choose the most preferred one. Then, the search will be guided to the neighborhood of the selected. The weights of selected solutions will be used to guide the optimization for finding the finial preferred region.

IV. MOEA/D

MOEA/D is a generic algorithm framework. A full analysis of MOEA/D is given in [11]. Here we only summarize main points. Basically it decomposes a multi-objective optimization problem into a number of different single objective optimization sub-problems and defines neighborhood relations among these sub-problems. Then a population-based method is used to optimize these sub-problems simultaneously. Each sub-problem is optimized by using information mainly from its neighboring sub-problems. The algorithm is described as following:

Input parameters: A multi-objective optimization problem, A stopping criterion, N sub-problems considered in MOEA/D, a set of weight vectors (being uniformly spread): $\lambda^1 \ldots \lambda^N$, and T as the neighborhood size for each weight vector.

Output: an External Population (*EP*) storing all non-dominated solutions)

Step 1) Initialization:

Step 1.1) Set EP = ∅;.

Step 1.2) Compute the Euclidean distances between any pair of weight vectors and then find out T closest weightvectors to each weight vector. For each i= $1, \ldots, N$, set

$B(i) = \{i_1, \ldots, i_T\}$, where $\lambda^{i1}, \ldots, \lambda^{iT}$ are T closest weight vectors to λ^i

Step 1.3) Generate an initial population $x^1, \ldots, x^N$ randomly or by a problem-specific method. Set $FV^i = F(x^i)$.

Step 1.4) Initialize $z = (z_1, \ldots, z_m)^T$ by a problem-specific method.

Step 2) Update:

For i = $1, \ldots, N$ do

Step 2.1) Reproduction: Randomly select two indexes k; l from $B(i)$, and then generate a new solution y from x^k and x^l by using genetic operators.

Step 2.2) Improvement: Apply a problem-specific repair/improvement heuristic y on to produce y' .

Step 2.3) Update of z: For each $j = 1, \ldots, m$ if $z_j < f_j(y')$

then set $z_j = f_j(y')$.

Step 2.4) Update of Neighboring Solutions: For each index $j \in B(i)$,if $g^{te}(y'|\lambda^j; z) < g^{te}(x^j|\lambda^j; z)$, then set $x^j = y'$ and $FV^j = F(y')$ (see Eq. 3 for g^{te} information)

Step 2.5) Update of EP:

Remove from EP all vectors dominated by $F(y')$.

Add $F(y')$ to EP if no vectors in EP dominate $F(y')$.

Step 3) Stopping Criteria: If stopping criteria is satisfied, then stop and output EP. Otherwise, go to **Step 2**.

Note that in **Step 2** the authors proposed to use a Tchebycheff approach for converting the problem of approximation of the Pareto Front into a number of scalar optimization problems in the form:

$$Minimize\ g^{te}(x|\lambda, z^*) = max_{1<i<m}\{\lambda_i f_i(x) \quad (3)$$

subject to $x \in \Omega$

where $z^* = (z_1^*, \ldots, z_m^*)^T$ is the ideal point (also known as the reference point).

This ideal point is initialized in ***Step 1.4*** by problem-specific method. It is updated in **Step 2.3**. The ideal point replaces x^j with y'or not when y' performs better than x^j with regard to the j^{th} sub-problem. For that reason, the ideal point has an important role in optimal process. We suggest modifying it with DM's preference information through interaction during process in the next section.

V. METHODOLOGY

We propose a method to Interactive for MOEA/D followingthe model in Fig. 3:

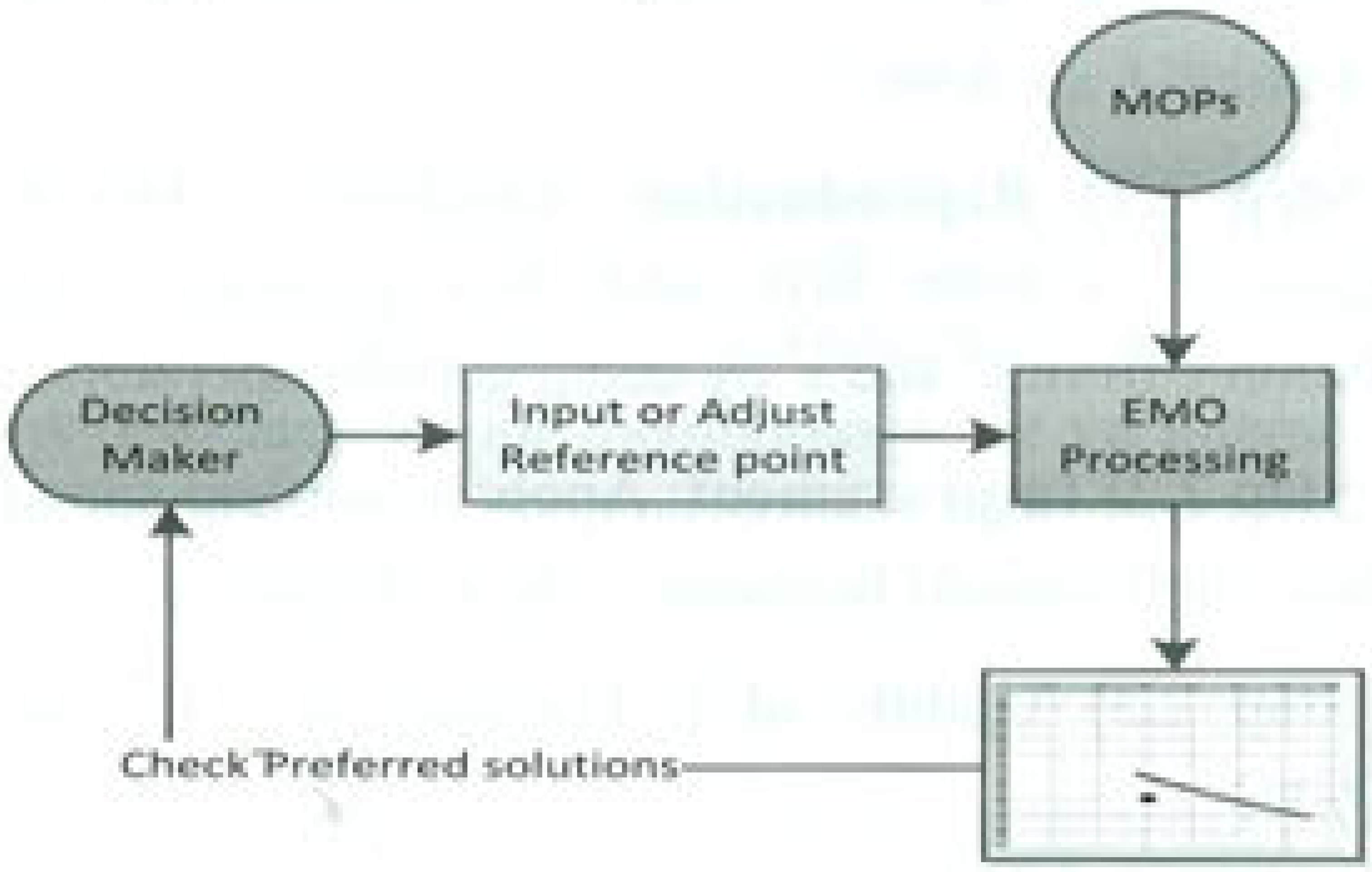

Figure 3. Model for MOEA/D interaction using a reference point.

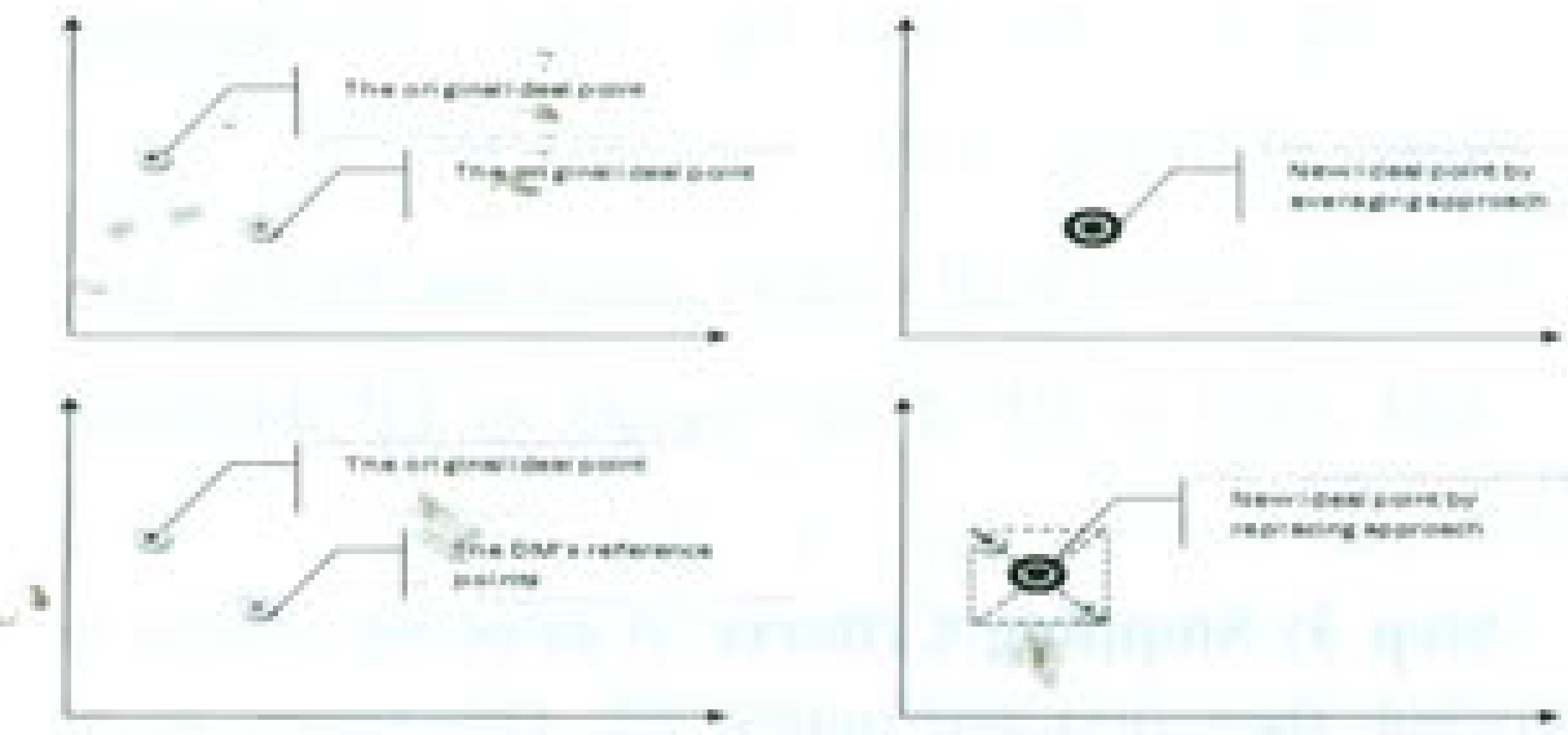

Figure 4. Two ways using reference point in Interactive with MOEA/D

In this model, DM has two actions during the Interaction process, the first one is *checking preferred solutions* in current generation, and second one is *inputting or adjusting the reference point* if DM is not satisfied. In more details, during the optimization process, DM is asked to input or to adjust a reference point in the objective space. After DM has specified a reference point in the objective space, a suitable set of Pareto optimal solutions is found corresponding to preference. If DM is not satisfied, he/she can specify other reference points.

In proposed method, DM gives a reference point in the objective space at each iteration. We use this point as the ideal point for MOEA/D by two ways:

Replace the original ideal point by DM's reference point. (*See Fig. 4 - the left graph*)

Generate new ideal point by averaging value between original ideal point and DM's reference point in objective space (*See Fig. 4 - the right graph*). With this combination, we hope that the system can provide DM more useful information of the optimal area. For example, at least DM can estimate how close the preferred area towards the optima one (being represented by the ideal point).

In the averaging approach, we can get new ideal point z_i^* values by the following formula:

$$z_i^* = \frac{z_i + r_i}{2} \qquad (4)$$

Here r_i is the value of i^{th} objective given by DM, z_i the value of the current ideal point for i^{th} objective.

We have an additional step call ***Interactive step*** before **Step 2.4** of MOEA/D, it does:

Ask DM to input or adjust a reference point in objective space.

Ask DM to choose a method for integration with ideal point (replace or combine).

Compute for a new ideal point, put new the ideal point to the optimal process.

For the pseudo-code of the interactive procedure, we use function names ***AddingDMReference*** to add new reference point that created from DM's reference points:

```
AddingDMReference()
{
if ( replace == true)
 {
     idealpoint[0] = DMidealpoint[0][0];
     idealpoint[1] = DMidealpoint[0][1];
 }
else
{
        idealpoint[0] = (idealpoint[0] +
        DMidealpoint[0][0])/2;
```

```
        idealpoint[1] = (idealpoint[1] +
        DMidealpoint[0][1])/2;

    }
    for(inti=0; i<pops; i++)
        update_reference(population[i].indiv);

    }
```

It is important to differentiate our method from the one proposed in [6]. It can be described as follows: For the method in [6], the uniformly spread weight vectors are obtained for decomposing the MOP. The process of interacting with DM will happen periodically. In that implementation, it happens once every *H* generations. At each interaction, P individuals are presented to DM. After estimating the utility function values of these individual solutions, the best solution *y** will be selected as the center of the new preferred region. Meanwhile, our proposal is not to choose the solutions, but ask DM to explicitly specify the point/region of preference in the space.

VI. CASE STUDIES AND DISCUSSIONS

A. Test functions

In our experiment, we used ZDT test problems in [12] with two objectives. The description of problems as follows:

ZDT1: two objectives, and convex POF:

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)\left[1 - \sqrt{\dfrac{x_1}{g(x)}}\right] \\ g(x) = 1 + 9\dfrac{\sum_{i=1}^{n} x_i}{n-1} \end{cases} \quad (5)$$

n = 30, $x_i \in [0; 1]$ with $i=1,\dots,n$;

ZDT2: two objectives, and non-convex POF.

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)\left[1 - \left(\dfrac{x_1}{g(x)}\right)^2\right] \\ g(x) = 1 + 9\dfrac{\sum_{i=1}^{n} x_i}{n-1} \end{cases} \quad (6)$$

n = 30, $x_i \in [0; 1]$;

ZDT3: two objectives, and convex and disconnected POF

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)\left[1 - \sqrt{\dfrac{x_1}{g(x)}} - \dfrac{x_1}{g(x)}\sin(10\pi x_1)\right] \\ g(x) = 1 + 9\dfrac{\sum_{i=1}^{n} x_i}{n-1} \end{cases} \quad (7)$$

n = 30, $x_i \in [0; 1]$;

ZDT4: two objectives, and convex POF.

$$f_1(x) = x_1$$

$$f_2(x) = g(x)\left[1 - \sqrt{\frac{x_1}{g(x)}} - \frac{x_1}{g(x)}\sin(10\pi x_1)\right]$$

$$g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}\left[x_1^2 - 10\cos(4\pi x_i)\right] \quad (8)$$

n = 10, $x_1 \in [0; 1]$; $x_i \in [-5; 5]$ with $i=2,\dots,n$.

ZDT5: two objectives, and non-convex, non-uniformly spaced POF.

$$\begin{cases} f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1) \\ f_2(x) = g(x)\left[1 - \left(\dfrac{f(x_1)}{g(x)}\right)^2\right] \\ g(x) = 1 + 9\left[\sum_{i=2}^{n}\dfrac{x_i}{n-1}\right]^{0.25} \end{cases} \quad (9)$$

n = 10; $x_i \in [0; 1]$

B. Results and Discussions

The common parameters for MOEA/D in our experiments were set with values given in Table 1.

Table 1. Parameters for moea/d in our experiments

Parameter	Value	Comments
Max. number of generations	300	Number of generation to get final feasible solutions
Neighborhood size	20	A parameter for MOEA/D

To understand the behavior of MOEA/D, we ran it and recorded the positions of the ideal points over time. In all five problems (See Fig. [5…7]), the ideal point was approximated and was approaching the origin as the time progressed.

For interaction, we setup the scenario as in Fig. 8 with two options of changing the original ideal point of MOEA/D. Given the area of preference as depicted in Fig. 9, 10, 11,12 and 13, we found the segments of POFs for each ZDT accordingly.

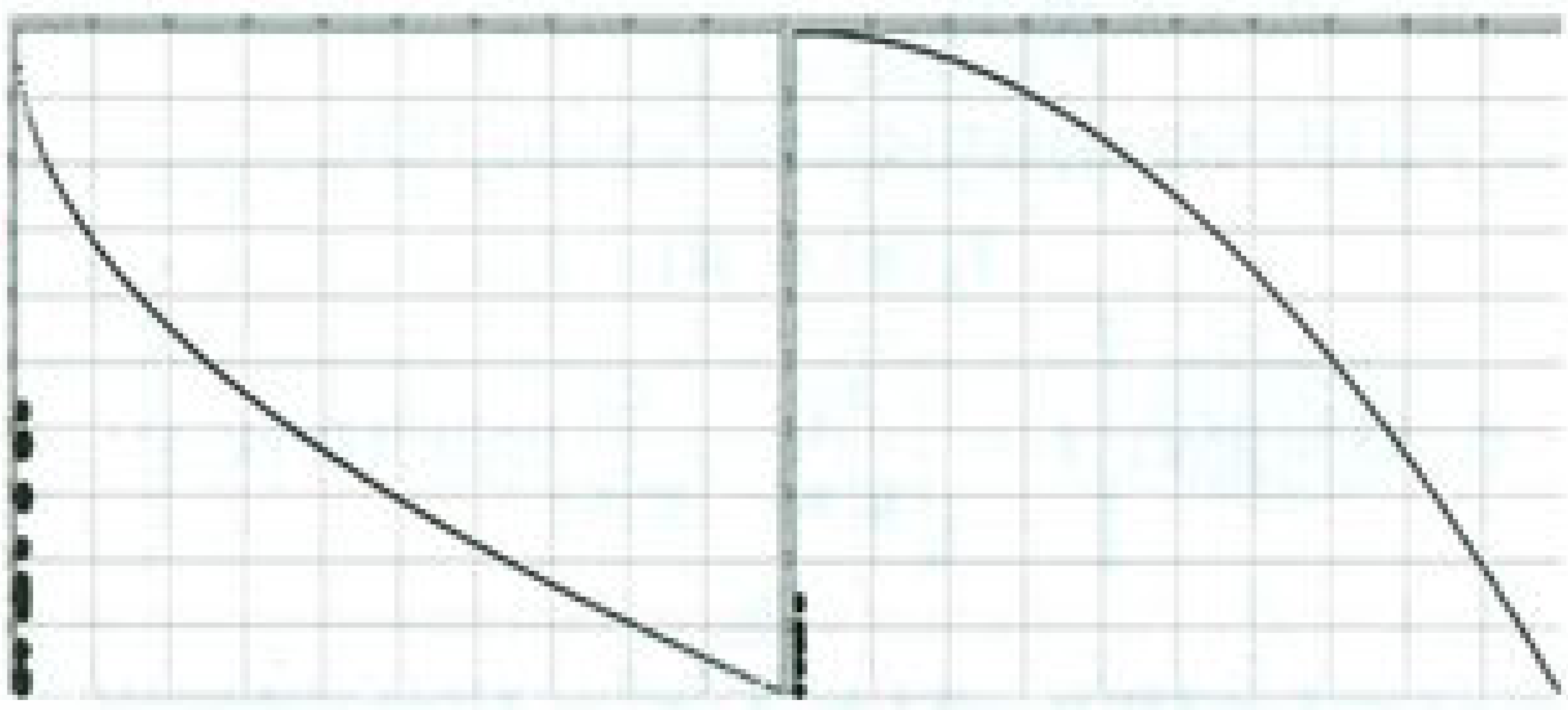

Figure 5. The ideal points are distributed on object space without interaction (ZDT1 - Left, ZDT2 - Right)

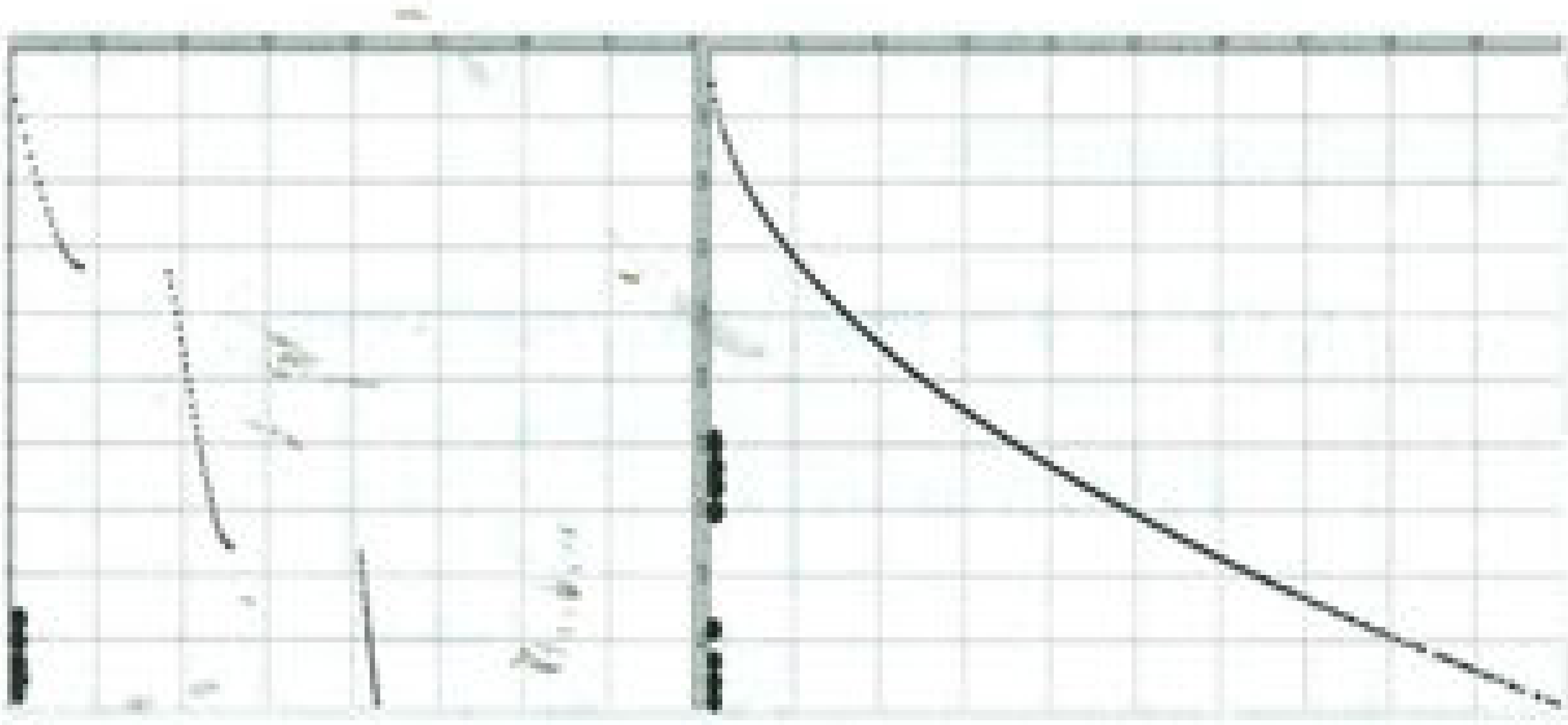

Figure 6. The ideal points are distributed on object space without interaction (ZDT3 – Left, ZDT4 – Right)

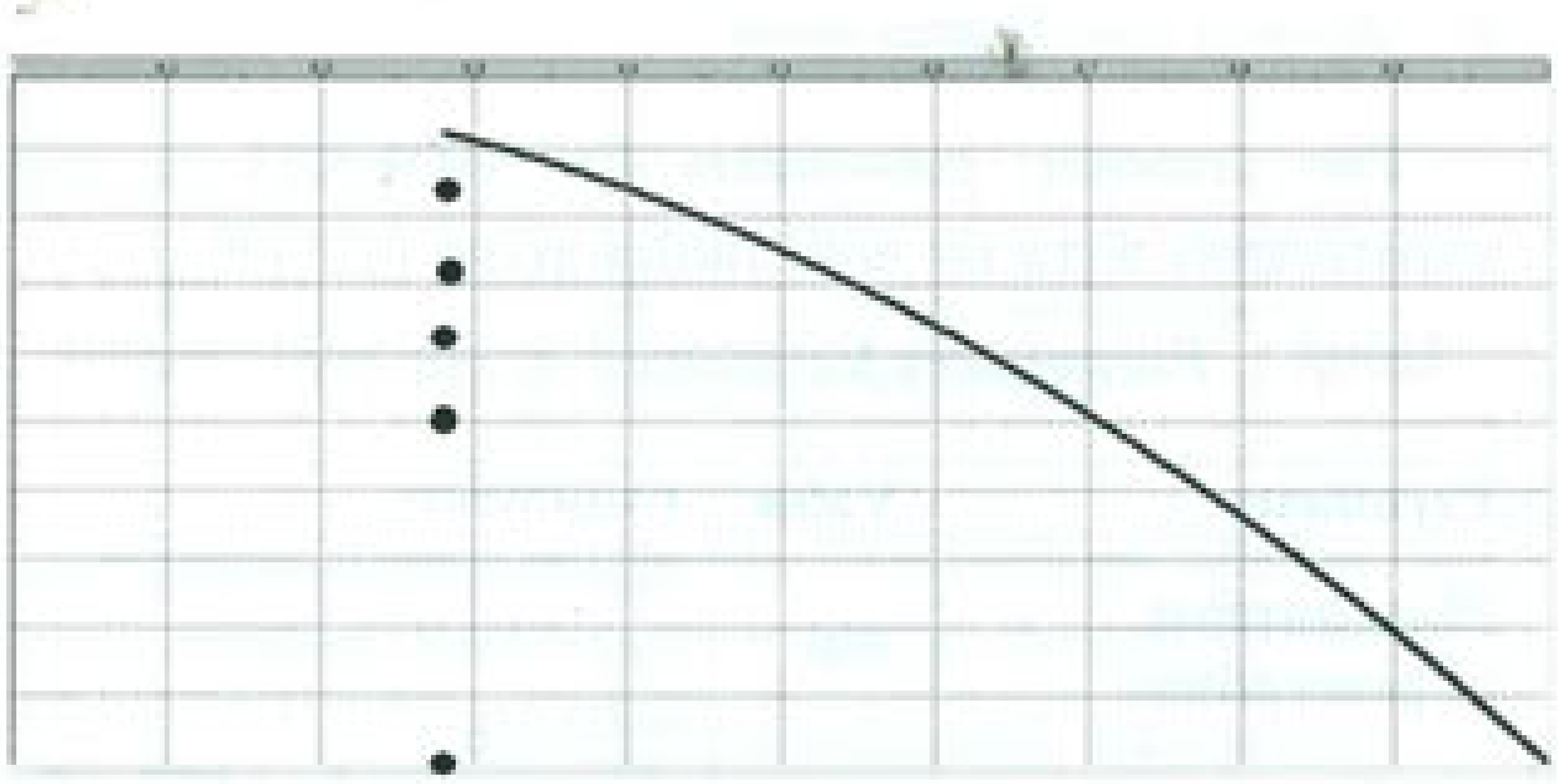

Figure 7. The ideal points are distributed on object space without interaction (ZDT5)

It was clear that if we completely replaced the ideal point by the one the segment of POFs was contracted towards the area of preference. This is somewhat lessened in the case of combining the current ideal point (with larger segment of POFs). We expect that in some difficult cases, combination of the current ideal point with DM's preference can guide the search towards the global optima.

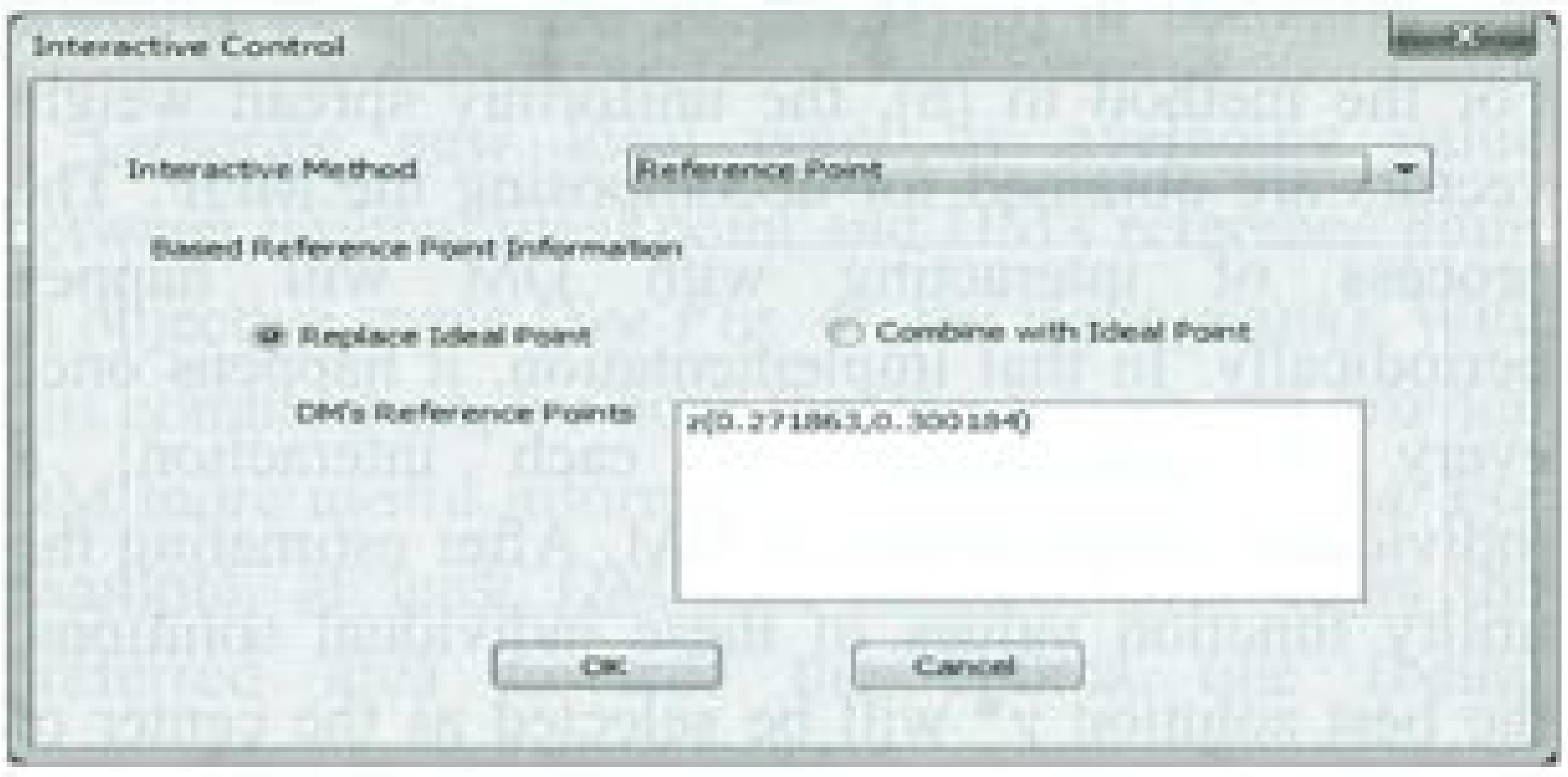

Figure 8. Visual screen for DM during optimal process

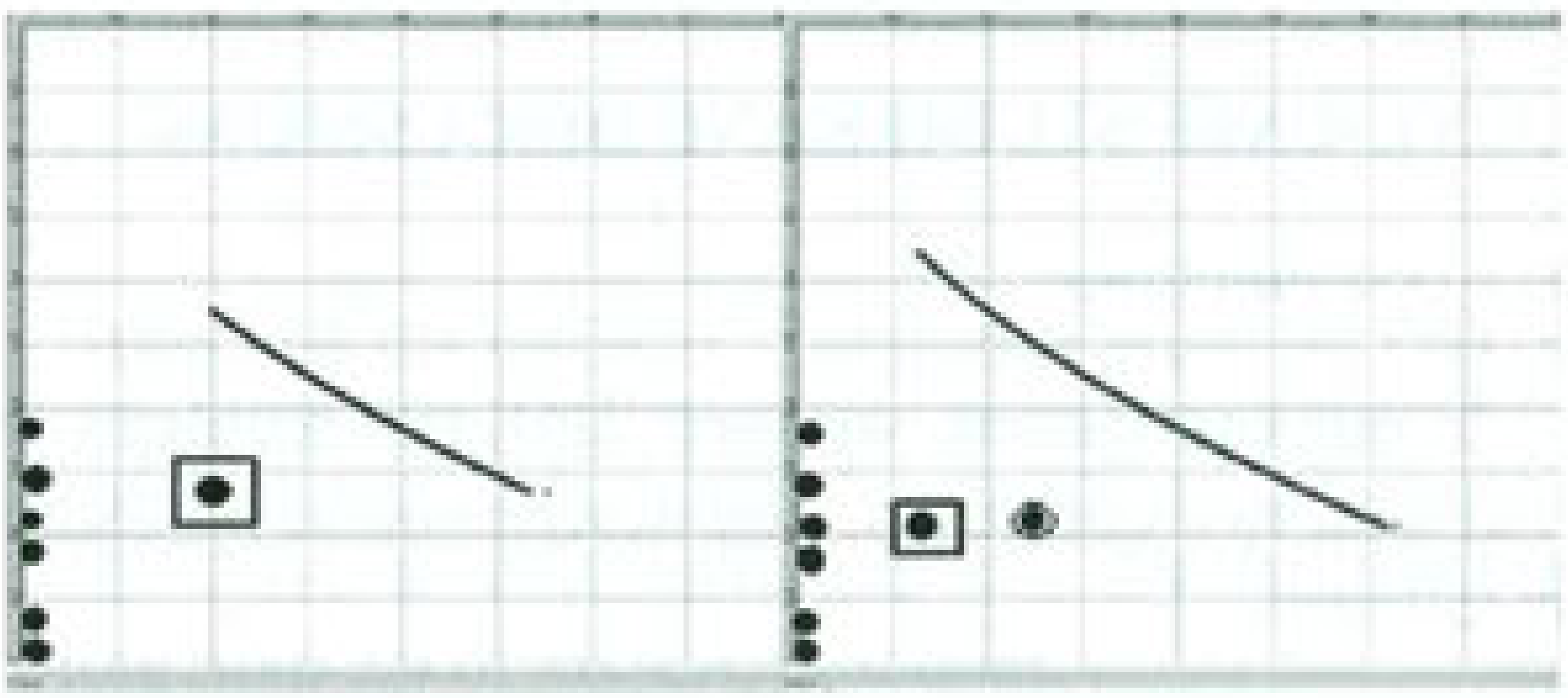

Figure 9. Test problem ZDT1. Two methods: replacing current Ideal point (Left) and combining (Right), new Ideal point in the rectangle bound, DM's reference point in the circle bound.

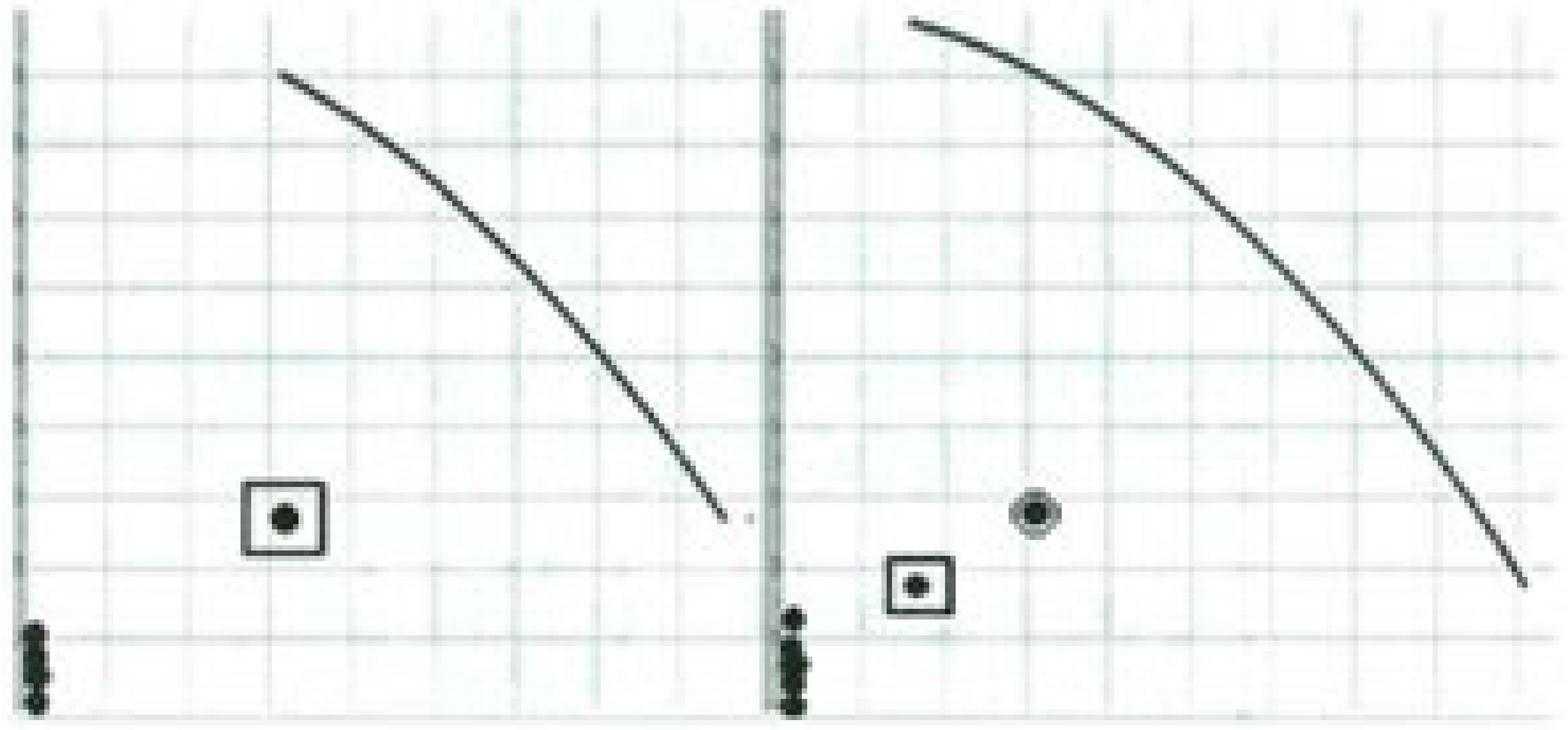

Figure 10. Test problem ZDT2.Two methods: replacing current Ideal point (Left) and combining (Right), new Ideal point in the rectangle bound, DM's reference point in the circle bound.

In summary, by using interactive method with MOEA/D, where the reference point is given by DM, The final solutions will be strongly converged to the DM's preferred region. It ensures diversity of population and principle of the MOEA/D. With the interactive method help DM to get the most preferred solutions.

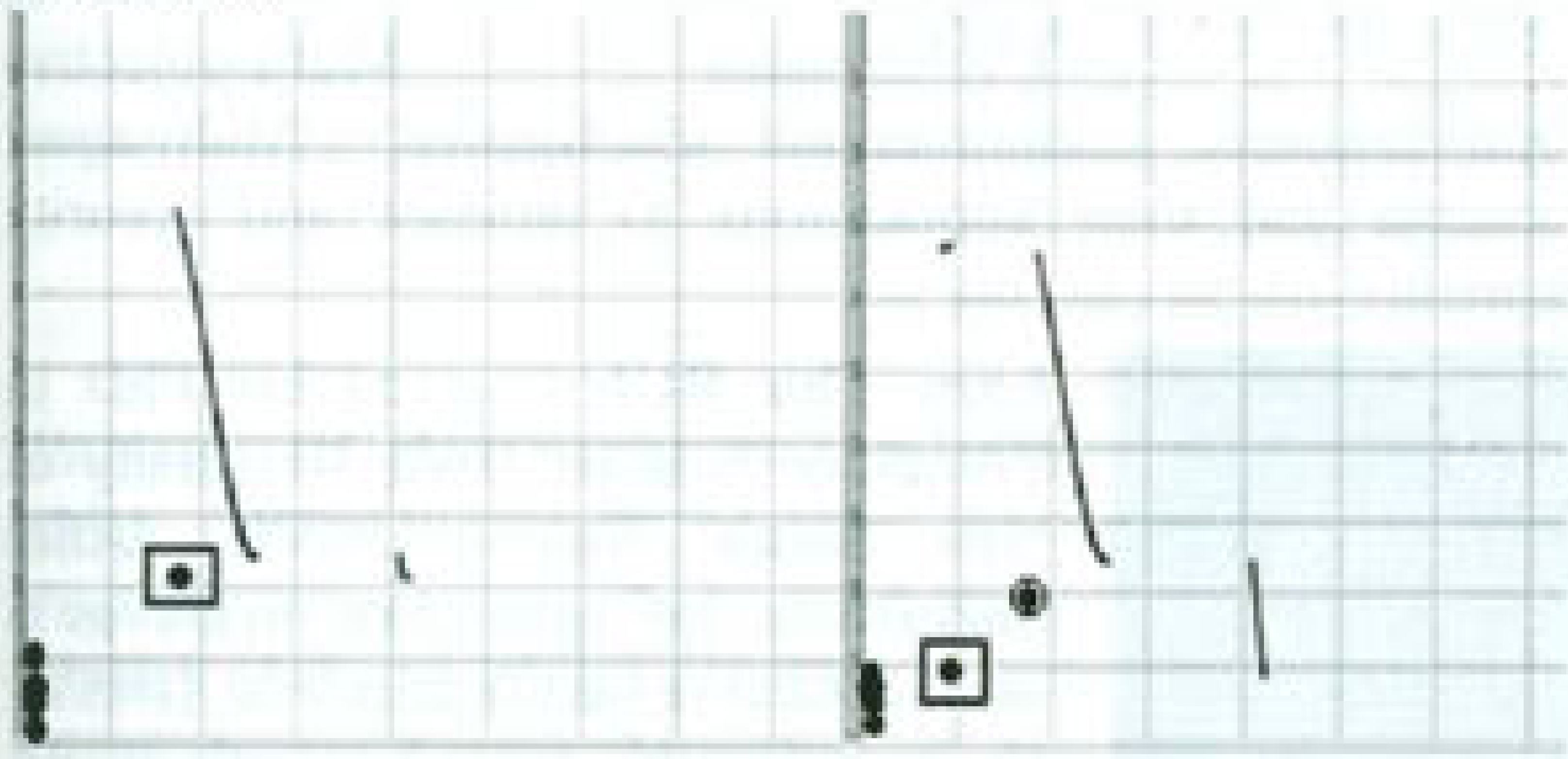

Figure 11. Test problem ZDT3. Two methods: replacing current Ideal point (Left) and combining (Right), new Ideal point in the rectangle bound, DM's reference point in the circle bound.

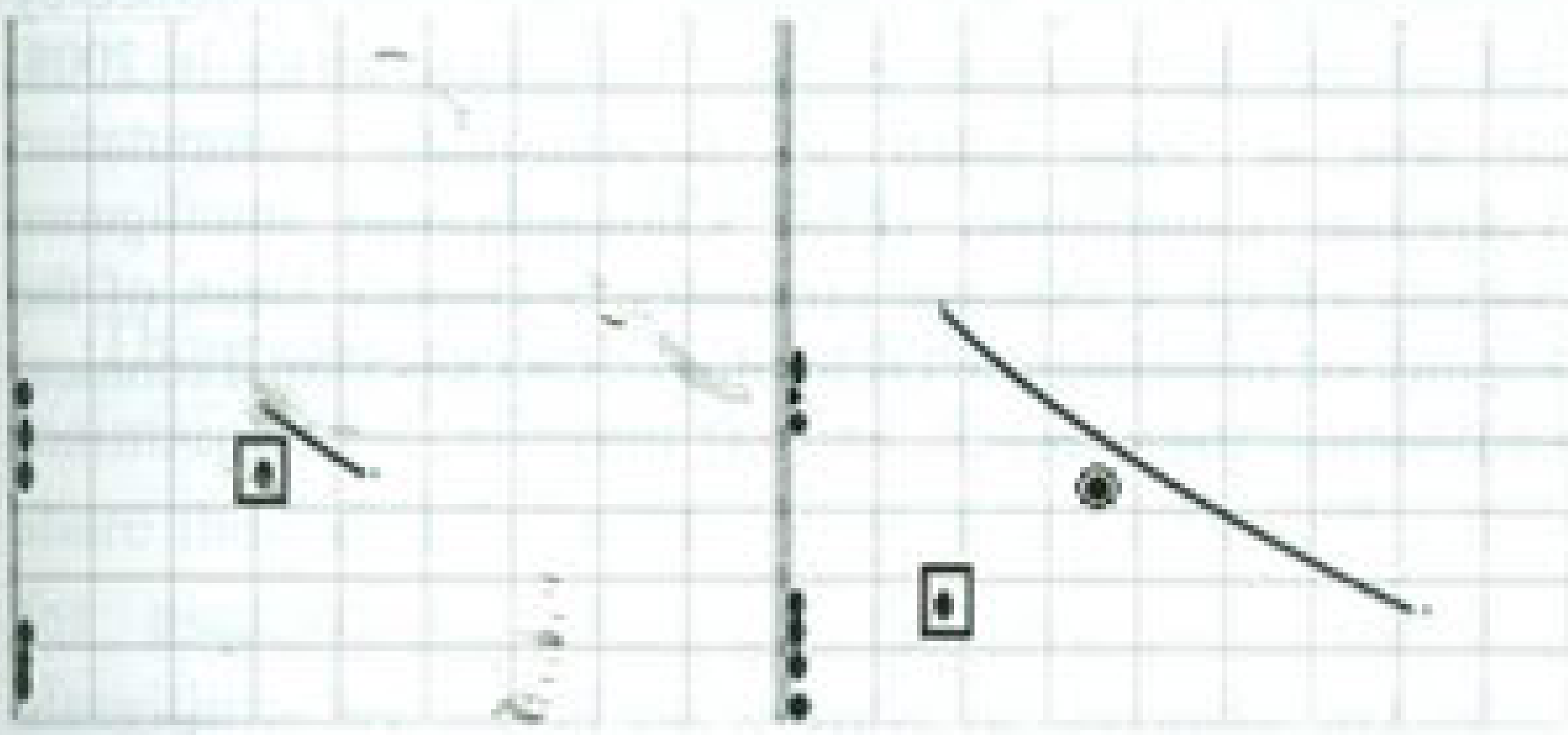

Figure 12. Test problem ZDT4.Two methods: replacing current Ideal point (Left) and combining (Right), new Ideal point in the rectangle bound, DM's reference point in the circle bound.

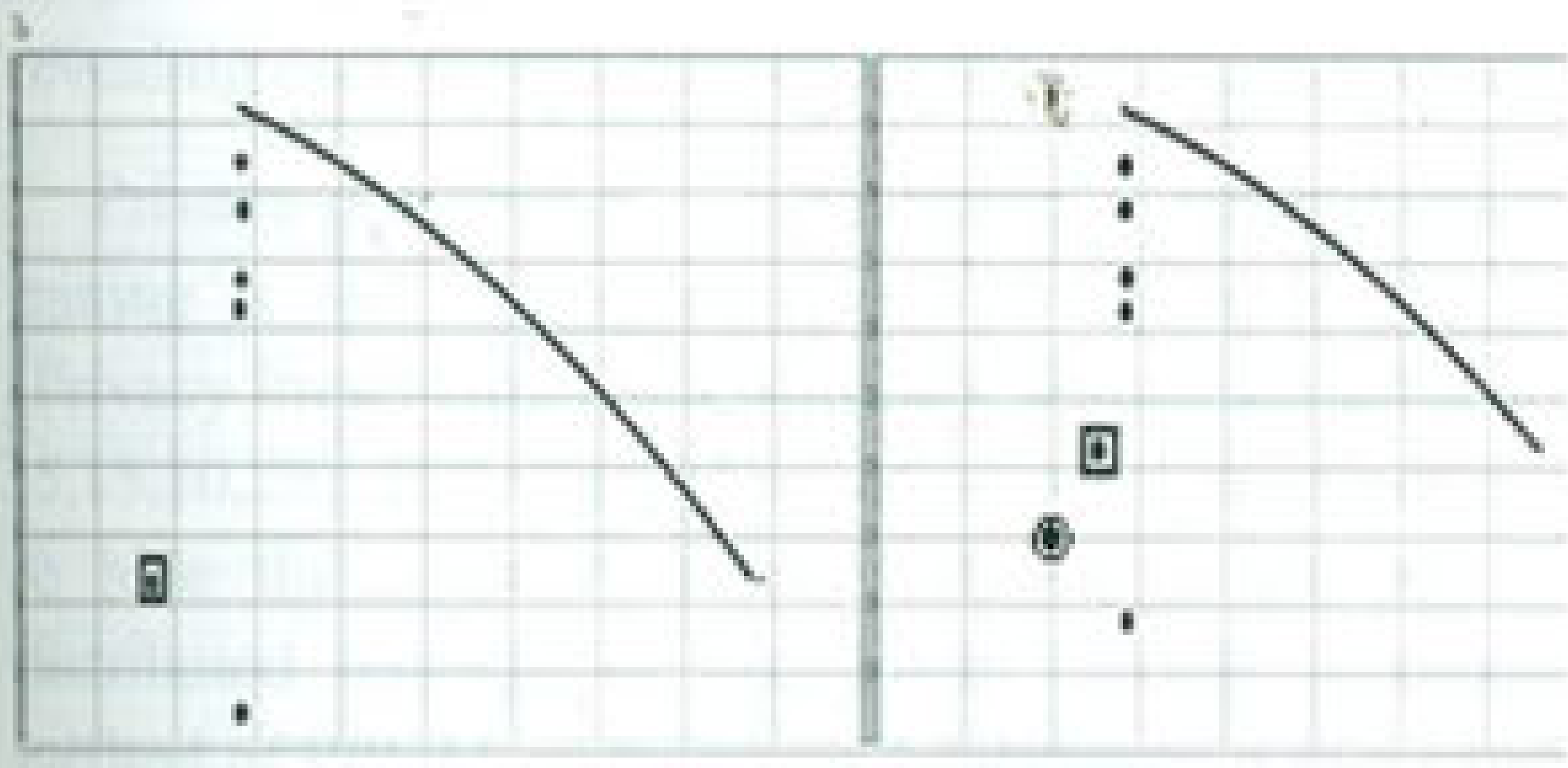

Figure 13. Test problem ZDT5. Two methods: replacing current Ideal point (Left) and combining (Right), new Ideal point in the rectangle bound DM's reference point in the circle bound.

VII. CONCLUSION

In this paper, we proposed an interactive method using a reference point with multi-objective optimization based on decomposition-based MOEA (MOEA/D). We used a reference point in objective space to represent for DM's preferred region. The reference point is used in optimal process by two ways: replace or combine the current ideal point at the loop. In all experiments, we used ZDT problems with two objectives, but the extension to higher-dimension problems is possible.

Through our experiments, we found that by our alternative method by both ways, we can get strongly convergence of final solutions to DM's preferred region in the objective space, while ensuring the diversity of population and principle of the MOEA/D.

ACKNOWLEDGMENT

We acknowledge the partial financial support from Vietnam's National Foundation for Science and Technology (Development Grant 102.01-2010.12). The authors also would like to thank the Vietnam Institute for Advance Study in Mathematics (VIASM) for their partial financial support.

REFERENCES

[1] Wierzbicki A. The use of reference objectives in multi-objective optimisation. MCDM theory and Application, Proceedings. No. 177 in Lecture notes in economics and mathematical systems, pages 468–486, 1980.

[2] K. Deb. Multi-objective optimization using evolutionary algorithms. John Wiley & Sons, New York, 2001.

[3] K. Deb and A. Kumar. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In GECCO '07, pages 781–788, 2007.

[4] K. Deb, A. Sinha, P. J. Korhonen, and J. Wallenius. An interactive evolutionary multi-objective optimization method based on progressively approximated value functions. Volume: 14, Issue 5 , IEEE Transactions on Evolutionary Computation, IEEE Press, pp 723 - 739 , 2010.

[5] K. Deb and J. Sundar. Reference point based multi-objective optimization using evolutionary algorithms. In GECCO '06: Proceedings of the 8th annual conference on Genetic and Evolutionary Computation, pages 635–642, New York, NY, USA, 2006. ACM Press.

[6] Maoguo Gong, Fang Liu, Wei Zhang, Licheng Jiao, and Qingfu Zhang. Interactive moea/d for multi-objective decision making. In GECCO' 2011, pages 721–728, 2011.

[7] Ed. Branke J., Deb K., Miettinen K., Slowinski R., and Berlin. Consideration of partial user preferences in evolutionary multi-objective optimization. multi-objective optimization: interactive and evolutionary approaches. OR Spectrum, 2008, Springer.

[8] Eskelinen Petri and Miettinen Kaisa. Trade-off analysis approach for interactive nonlinear multiobjective optimization. In OR Spectrum, pages 1–14, 2011, online first, Springer.

[9] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina. A preference based evolutionary algorithm for multi-objective optimization. Pages 411–436, Evolutionary Computation archiveVolume 17 Issue 3, Fall 2009, MIT Press Cambridge, MA, USA.

[10] Belton V., Branke J., Eskelinen P., Greco S., Molina J., Ruiz F., and Slowinski. Interactive multi-objective optimization from a learning perspective. multi-objective optimization: interactive and evolutionary approaches. OR Spectrum, 2008.

[11] Q. F. Zhang and H. Li. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Trans. Evolutionary Computation 11(6): 712-731, 2007.

[12] E. Zitzler, L. Thiele, and K. Deb. Comparision of multiobjective evolutionary algorithms: Emprical results. Evolutionary Computation, 8(1):173–195, 2000.

AUTHORS' BIOGRAPHIES

Long NGUYEN has his B.Sc (2000) and M.Tech (2007) degree from Le Quy Don Technical University. Currently he is a PhD Candidate at the Faculty of Information Technology, Le Quy Don Technical University. His research interests are in the areas of Evolutionary Multi-objective Optimizations, Computational Intelligence, Multi-agent systems and their applications to defence and security domain.

Lam Thu BUI received his PhD in Computer Science from University of New South Wales (UNSW). After graduation, he spent two years as a postdoctoral research fellow at UNSW. He is currently doing research in the field of evolutionary computation, in particular Evolutionary Multi-Objective Optimization. He is the co-editor of the book "Multi-objective Optimization in Computational Intelligence: Theory and Practice", which was published by IGI Global, Information Science Reference series in 2008. Dr Bui is Vice Chair of Evolutionary Computation Technical Committee, IEEE Computational Intelligence Society (IEEE CIS), and currently is Deputy Dean of the Faculty of Information Technology, Le Quy Don Technical University, Vietnam