A Multi-Point Interactive Method For Multi-Objective Evolutionary Algorithms

Long Nguyen
Faculty of Information Technology
Le Quy Don Technical University
Hanoi, Vietnam
Email: longit76@gmail.com

Lam Thu Bui
Faculty of Information Technology
Le Quy Don Technical University
Hanoi, Vietnam
Email: lam.bui07@gmail.com

Abstract—**Many real-world optimization problems have more than one objective (and these objectives are often conflicting). In most cases, there is no single solution being optimized with regards to all objectives. Deal with such problems, Multi-Objective Evolutionary Algorithms (MOEAs) have shown a great potential. There has been a popular trend in getting suitable solutions and increasing the convergence of MOEAs, that is consideration of Decision Maker (DM) during the optimization process (interacting with DM) for checking, analyzing the results and giving the preference.**

In this paper, we propose an interactive method allowing DM to specify a set of reference points. It used a generic algorithm framework of MOEA/D, a widely-used and decomposition-based MOEA for demonstration of concept. Basically MOEA/D 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. In MOEA/D an ideal point is used to choose neighbored solutions for each run. Instead of using a single point, we introduce an alternative to the set of reference points. There are several way to take into account the information of the region specified by the set of reference points; here we used the mean of this set (or we call the combined point). The combined point which represents for the set of reference points from DM is used either to replace or adjust the current ideal point obtained by MOEA/D. We carried out a case study on several test problems and obtained quite good results.

Keywords-**Multi-point; Interactive; Reference point; MOEA/D.**

I. INTRODUCTION

When solving multi-objective optimization problems (MOPs), we need to simultaneously optimize several objective functions [2]. As a result, we usually obtain *trade-offs*, 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 process by indicating his/her preference information over the set of solutions until DM satisfies or prefers to stop the process[9]. An interesting feature of interactive methods is that during the optimal process DM is able to learn about the underlying problem as well as his/her own preference. To date, many interactive techniques have been proposed [3], [4], [5], [6], [7], [8], [10] for solving MOPs. It is worthwhile to note that the aim of the interactive method is to find most suitable solution in several conflicting

objectives regarding the DM's preference. It requires a mechanism to support DM in formulating her/his preferences and identifying preferred solutions in the set of Pareto optimal solutions.

In this paper, we introduce an interactive method integrating with MOEA/D [11], a decomposition-based multi-objective evolutionary algorithm. With this proposal, we allow DM to specify multiple reference points at each interaction. In this way, DM has more flexibility to epress his preference. Among several methods for taking set information, we propose to use the mean (the average point from the set). The ideal point (or the reference point) requested by MOEA/D is replaced by the average point from DM's set of reference points. We hypothesise that a multi point approach is a good way to express DM's preferences, since specifying a set of reference points is considerably convenient for DM. It first consists of desirable objective values, so-called *aspiration* levels (for each objective). After DM has specified a set of reference points, a set of Pareto optimal solutions is found that best corresponds to them. If DM is not satisfied, he/she can specify other reference points. In our research, we adjust the ideal point by either replacing the ideal point by the aggregated reference point being resulted from DM's set of reference points; or combining the ideal point and the aggregated reference point.

In the remainder of the paper, section II briefly describes about multi-objective optimization interactive method using reference point. Thereafter, in Section III we show some related interactive MOEAs methods. In section IV we have description for MOEA/D, section V we propose our methodology for an interactive with MOEA/D. Section VI presents simulation results and discussion. Finally, the conclusion of this paper is outlined in section VII.

II. REFERENCE POINT INTERACTIVE APPROACH

The reference point interactive method is suggested by Wierzbicki[1], this method is known is classical reference point approach. The idea of the method to control the search by reference points using *achievement functions*. Here the achievement function is constructed in such a way that if the reference point is dominated, the optimization will advance past the reference point to a non-dominated solution.A reference point z^* is given for an M-objective optimization problem of minimizing

$(f_1(x), \ldots, f_k(x))$ with $x \in S$. Then single-objective optimization one as following:

$minimize$

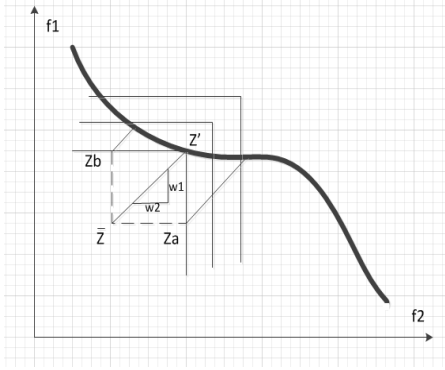$$max_{i=1}^{M}[w_i(f_i(x) - z_i^*)] \qquad (1)$$

subject to $x \in S$.

Figure 1. Altering the reference point, Here Z_A, Z_B are reference points, w is chosen weight vector used for scalarizing the objectives

The algorithm for this method is described in five following steps:

Step 1: Present information to the DM. Set h=1

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

Step 3: Minimize achievement function. Present z^h to the DM

Step 4: Calculate k other solutions with reference points
$\overline{z}(i) = \overline{z}^h + d^h e^i$ where $d^h = ||z_*^h - z^h||$ and e_i is the ith unit vector

Step 5: If the DM can select the final solution, stop. Otherwise, ask the DM to specify z_*^{h+1}. Set $h = h + 1$ and go to **Step 3**.
Here h is the number that DM specifies a reference point during process. By the way of using the series of reference points, DM actually tries to evaluate the region of Pareto Optimality, instead of one particular Pareto-optimal point. However DM usually deals with two situations:

1) The reference point is feasible and not a Pareto-optimal solution, the DM is interested in knowing solutions which are Pareto-optimal ones and near the reference point.
2) The DM finds Pareto-optimal solutions which is near supplied reference point.

III. RELATED INTERACTIVE MOEAs

In this section, we summarize several typical works on this area. In [5], authors proposed an Interactive MOEA using a concept of the reference point and finding a set of preferred Pareto optimal solutions near the regions of interest to a DM. The authors suggest two approaches:

The first is to modify NSGA-II for effectively solving 10-objective. The other is to use 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 [3]. In this method supplies one or more reference directions in the objective space. The proposed method is exploited to find a set of efficient solutions corresponding to a number of reference points known as the reference direction. 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.

The authors proposed in [8] a trade-off analysis tool that was used to offer the DM a way to analyze solution candidates. The ideas proposed here are directed to users of both classification and reference point based methods. The motivation here is that DM in certain cases miss additional local trade-off information so that they could get to know how values of objectives are changing, in other words, in which directions to direct the solution process so that they could avoid trial-and-error, that is, specify some preference information so that more preferred solutions will be generated.

An interactive version of the decomposition based multi-objective evolutionary algorithm (iMOEA/D) is also discussed in [6]. For iMOEA/D, MOP in question is converted into a number of scalar optimization problems by the Tchebycheff approach with even spread weight vectors. During optimization process, DM is asked to compare some current solutions and select ones, which are please them most at each interactive stage. 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

We propose an interactive method with MOEA/D using reference method. MOEA/D is a generic algorithm framework. 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[11] as following:
Input:
MOP (1);
a stopping criterion;
N: the number of the subproblems considered in MOEA/D;
a uniform spread of weight vectors: $\lambda^1 \ldots \lambda^N$
T: the number of the weight vectors in the neighborhood

of each weight vector.

Output:EP.

Step 1) *Initialization*:

Step 1.1) Set $EP = \emptyset$.

Step 1.2) Compute the Euclidean distances between any two weight vectors and then work out the T closest weight vectors to each weight vector. For each $i = 1, \ldots N$, set $B(i) = \{i_1 \ldots _1_T\}$, , where $\lambda^{i1} \ldots \lambda^{iT}$ are the Tclosest weight vectors to λ^i

Step 1.3) Generate an initial population $x^1,...,x^N$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 $g^{te}(y'|\lambda^j, z) \leq g^{te}(x^j|\lambda^j, z)$, then set $x^j = y'$ and $FV^j = F(y')$.

Step 2.5) *Update of* EP :

Remove from EP all the 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**.

In our experiments, we use Tchebycheff approach for converting the problem of approximation of the Pareto Front into a number of scalar optimization problems in the form:

Minimize $g^{te}(x|, z^*) = \max_{1<i<m}\{_i \; f_i(x) - z_i^*|\}$ subject to x$\in \Omega$ where $z^* = (z_1^*, \ldots, z_m^*)^T$ is the ideal point. An ideal point z is used during genetic process, it also known as a 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 jth sub-problem. For that reason, the ideal point has an important role in optimal process. We suggest to modify it with DM's preference information through interaction during process in the next section

V. METHODOLOGY

In our proposed method, we use multi reference points instead of using a single point for following reasons:

1) In some cases, DM does not have an exactly point of preference. So it is better to give them a facility for defining the region of interest.

2) The conventional interactive methods require DM giving several reference points via many iterations. Sometimes, they might be close in a region. Hence, it might be convenient for them to give these points at an one-go.
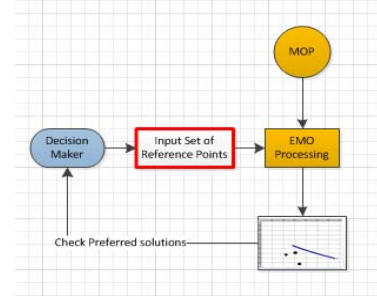

Figure 2.   Multi-point Interactive for EMO model.

In the method, DM gives a set of reference points in the objective space at the number of generations. A point is aggregated from the set of reference points and it is used as the ideal point by two ways:
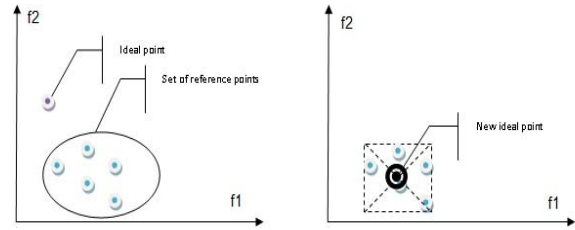

Figure 3.   Replace ideal point by aggregated point.

The first way we get an aggregated point from set of reference points by the average method. The we replace current ideal point by this point (see Fig.3). In this way we get new ideal point as below:

$$z_i^* = \frac{min(r_i^1, r_i^2, ..., r_i^K) + max(r_i^1, r_i^2, ..., r_i^K)}{2}$$

Here r_i is the prefered value of ith objective given by DM. K is number of reference points, $i = \{1, \ldots, K\}$
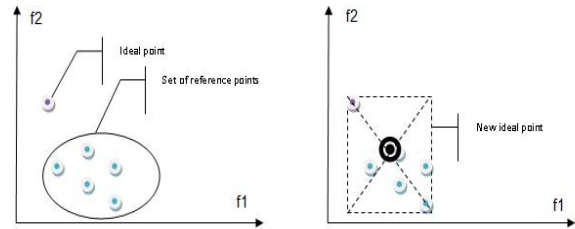

Figure 4.   Combine ideal point with aggregated point.

The second way, instead of replace ideal point with new one we combine aggregated point from set of reference points with current ideal point by average method (see Fig.4). The new ideal point is calculated as following:

$$z_i^* = \frac{min(r_i^1, r_i^2, \ldots, r_i^K, z_i) + max(r_i^1, r_i^2, \ldots, r_i^K, z_i)}{2}$$

Here z_i the best value of the current ideal point for i^{th} objective. We design an Interactive method with MOEA/D is described below: We have an additional step call ***Interactive step*** before Step 2.4 it does:
We use function names $addingDMReference$ follow to

- Ask DM like to interactive at current loop? If Yes go next, otherwise go to step 2.4.
- Ask DM to input or adjust reference points in objective space.
- Ask DM to choice 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.

add new reference point that created from DM's reference points:

```
void addingDMReference()
{
    double max1, min1, max2, min2;

    if ( replace == true)
    {
    if ( nDMpts ==1)
    {
    idealpoint[0] = DMidealpoint[0][0];
    idealpoint[1] = DMidealpoint[0][1];
    }
    else
    {
    max1= -MAXINT;
    max2 = -MAXINT;
    min1= MAXINT;
    min2 = MAXINT;
    for ( int j =0; j < nDMpts; j++)
    {
        if ( DMidealpoint[j][0] > max1)
           max1 = DMidealpoint[j][0];
        if ( DMidealpoint[j][1] > max2)
           max2 = DMidealpoint[j][1];
        if ( DMidealpoint[j][0] < min1)
           min1 = DMidealpoint[j][0];
        if ( DMidealpoint[j][1] < min2)
           min2 = DMidealpoint[j][1];
    }
    idealpoint[0] = (min1+ max1)/2;
    idealpoint[1] = (min2+ max2)/2;
    }
        }else
    {
    if ( nDMpts ==1)
    {
    idealpoint[0] = (idealpoint[0] + DMidealpoint[0][0])/2;
    idealpoint[1] = (idealpoint[1] + DMidealpoint[0][1])/2;
        }else
        {
    max1= -MAXINT;
    max2 = -MAXINT;
    min1= MAXINT;
    min2 = MAXINT;
        for ( int j =0; j < nDMpts; j++)
        {
        if ( DMidealpoint[j][0] > max1)
           max1 = DMidealpoint[j][0];
if ( DMidealpoint[j][1] > max2)
    max2 = DMidealpoint[j][1];
if ( DMidealpoint[j][0] < min1)
    min1 = DMidealpoint[j][0];
if ( DMidealpoint[j][1] < min2)
    min2 = DMidealpoint[j][1];
}
if ( idealpoint[0] > max1)
    max1 = idealpoint[0];
if ( idealpoint[0] < min1)
    min1 = idealpoint[0];
if ( idealpoint[1] > max2)
    max2 = idealpoint[1];
```

```
if ( idealpoint[1] < min2)
    min2 = idealpoint[1];
idealpoint[0] = (min1+ max1)/2;
idealpoint[1] = (min2+ max2)/2
        }
    }
}
```

The difference between our method and the proposed one in [6] is that: In [6], the uniformly spread weight vectors are obtained for decomposing the MOP. The process of interacting with DM will happen periodically. In his/her 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 new preference. Method [6]requires DM to estimate the utility function values of P individuals that being shown to DM. DM need to have experience for the problem to do the estimation. Our proposed method is quite simple for DM to determine their preferred region in objective space through inputting or adjusting a set of reference points for each interactive. Instead of estimation the utility function values of individuals, our method allows DM to express the preferred region through inputting or modifying set of reference points in objective space so that way is quite simply and more visually.

VI. CASE STUDIES AND DISCUSSION

A. Test functions

In our experiments, we use ZDT Test problems designed by Zitzler, Deb and Thiele [12]with two objectives:

- ZDT1: MOP with two objectives, convex.

  $\mathbf{f_1}(\mathbf{x}) = \mathbf{x_1}$

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

  $\mathbf{g}(\mathbf{x}) = \mathbf{1} + \mathbf{9}\frac{\sum_{i=1}^{n}\mathbf{x_i}}{\mathbf{n-1}}$

  $n = 30$, variables bound: $[0, 1], x \in [0, 1]$, $x_i = 0, i = 2, \ldots, n$.

- ZDT2: MOP with two objectives, non-convex.

  $\mathbf{f_1}(\mathbf{x}) = \mathbf{x_1}$

  $\mathbf{f_2}(\mathbf{x}) = \mathbf{g}(\mathbf{x})\left[\mathbf{1} - \left(\frac{\mathbf{x_1}}{\mathbf{g}(\mathbf{x})}\right)^{\mathbf{2}}\right]$

  $\mathbf{g}(\mathbf{x}) = \mathbf{1} + \mathbf{9}\frac{\sum_{i=1}^{n}\mathbf{x_i}}{\mathbf{n-1}}$

  $n = 30$, variables bound: $[0, 1], x \in [0, 1]$, $x_i = 0, i = 2, \ldots, n$.

- ZDT3: MOP with two objectives, convex and disconnected.

  $\mathbf{f_1}(\mathbf{x}) = \mathbf{x_1}$

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

  $\mathbf{g}(\mathbf{x}) = \mathbf{1} + \mathbf{9}\frac{\sum_{i=1}^{n}\mathbf{x_i}}{\mathbf{n-1}}$

  $n = 30$, variables bound: $[0, 1], x \in [0, 1]$, $x_i = 0, i = 2, \ldots, n$.

- ZDT4: MOP with two objectives, non-convex.

  $\mathbf{f_1}(\mathbf{x}) = \mathbf{x_1}$

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

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

$n = 10$, variables bound: $x_1 \in [0, 1], x_i \in [-5, 5]$.

- ZDT6: MOP with two objectives, Non-convex,Non-uniformly spaced.

$$\mathbf{f_1\,(x)} = \mathbf{1} - \mathbf{e}xp\,(\mathbf{-4x_1})\,\mathbf{sin^6(6\pi x_1)}$$

$$\mathbf{f_2\,(x)} = \mathbf{g(x)} \left[\mathbf{1} - \left(\frac{\mathbf{f(x_1)}}{\mathbf{g(x)}} \right)^{\mathbf{2}} \right]$$

$$\mathbf{g\,(x)} = \mathbf{1} + \mathbf{9} \left[\sum_{\mathbf{i=2}}^{\mathbf{n}} \frac{\mathbf{x_i}}{\mathbf{n-1}} \right]^{\mathbf{0.25}}$$

$n = 10$, variables bound: $[0, 1], x \in [0, 1]$,
$x_i = 0, i = 2, \dots, n$.

B. Results and Discussion

In MOEA/D the Ideal points for each problems are distributed during the process are show following figures (See Fig 5,6,7).
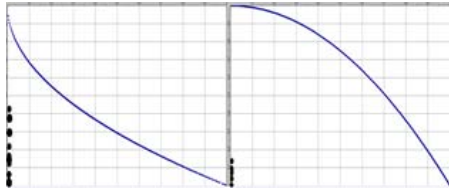

Figure 5. The ideal points are distributed on object space without interaction (ZDT1 on the left side, ZDT2 on the right side).
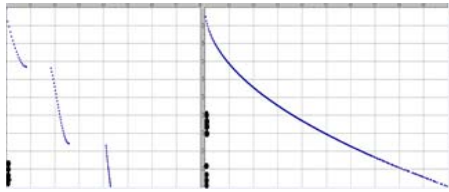

Figure 6. The ideal points are distributed on object space without interaction (ZDT3 on the left side, ZDT4 on the right side).
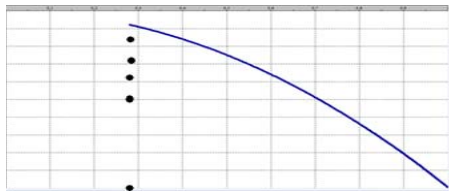

Figure 7. The ideal points are distributed on object space without interaction ZDT6.

During optimal process DM is asked in the visual screen. See Fig.8.

In TABLE I, we show the generation needed to get almost solutions in DM's preferred region in 3rd column. And comments are in 4th column. Through experiments with ZDT Test functions, we can see some features of the Interactive method:

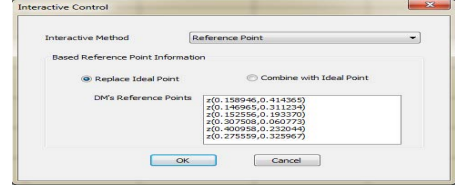1) Final solutions are strongly converged to region that DM prefers.

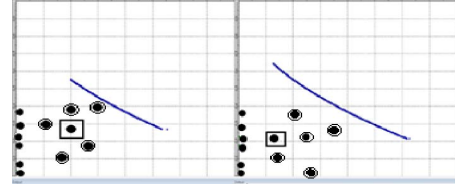Figure 8. Visual screen for DM during optimal process.

Figure 9. Test problem ZDT1:two methods: replacing current Ideal point (left side) and combining (right side), reference points are in circles, new Ideal point in the rectangle bound.
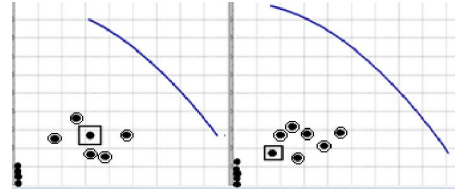

Figure 10. Test problem ZDT2:two methods: replacing current Ideal point (left side) and combining (right side),reference points are in circles, new Ideal point in the rectangle bound.
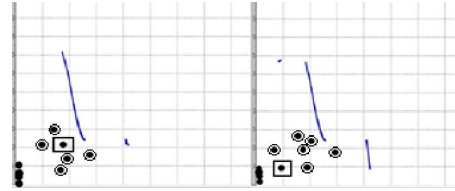

Figure 11. Test problem ZDT3:two methods: replacing current Ideal point (left side) and combining (right side), reference points are in circles, new Ideal point in the rectangle bound.
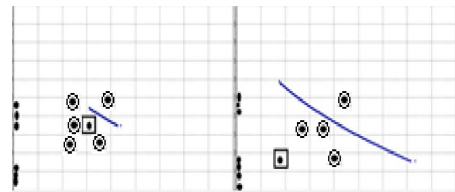

Figure 12. Test problem ZDT4:two methods: replacing current Ideal point (left side) and combining (right side), reference points are in circles, new Ideal point in the rectangle bound.

2) Some special cases PF's shape may be changed (such as ZDT3) with strong convergence of final solutions (see Fig.11).

By using an interactive method with MOEA/D, where reference points are 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

Table I
SUMMARY FOR THE EXPERIMENTS

| Problems	Points	Generation	Comments
ZDT1	6;5	38	With convex Pareto front, after interaction with DM, the final solutions are strongly converged to DM's preferred region. See Fig.9
ZDT2	6;6	53	With non-convex Pareto front, the final solutions are strongly converged to DM's preferred region.See Fig.10.
ZDT3	6;6	81	With convex and disconnected Pareto front, the final solutions not only are strongly converged to DM's preferred region, and they are also be distributed in other space in object space. See Fig.11.
ZDT4	6;4	46	With convex Pareto front, the final solutions are strongly converged to DM's preferred region. See Fig.12.
ZDT6	7;6	35	With non-convex,non-uniformly spaced, the final solutions are strongly converged to DM's preferred region in objective space. See Fig.13
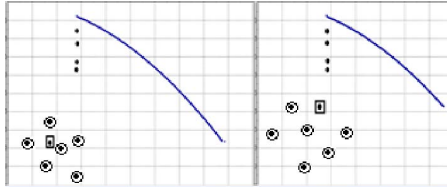
Figure 13. Test problem ZDT6:two methods: replacing current Ideal point (left side) and combining (right side), reference points are in circles, new Ideal point in the rectangle bound.

the most preferred solutions.

Our experiments were done at Software Technology Lab of Faculty of Software Technology, Le Quy Don Technical University.

VII. CONCLUSION

In this paper, we proposed an interactive method using multi reference points with multi-objective optimization based on decomposition-based MOEA (MOEA/D). In our alternative method we use a set of reference points in objective space to represent for DM's preferred region. The aggregated point form set of reference points is used in optimal process by two ways: replace or combine the current ideal point at the loop. In our experiments we use ZDTs problems with two objectives. Through our experiments, we found that by our alternative method by both ways, we can get strong convergence of final solutions to DM's preferred region in the objective space. But it ensures the diversity of population and principle of the MOEA/D.

ACKNOWLEDGMENTS

REFERENCES

[1] Wierzbicki A. The use of reference objectives in multi-objective optimisation. *MCDMtheory 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. 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.

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

[9] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina. A preference based evolutionary algorithm for multi-objective optimization. pages 411–436, 2009.

[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 multi-objective evolutionary algorithm based on decomposition. 2007.

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