International Journal of Computational Intelligence Research.
ISSN 0973-1873 Vol.2, No.3 (2006), pp. 273–286

Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms

Kalyanmoy Deb, J. Sundar, Udaya Bhaskara Rao N. and Shamik Chaudhuri

Department of Mechanical Engineering,
Indian Institute of Technology Kanpur,
PIN 208016, India
{*deb,jsundar,udaya,shamik*}*@iitk.ac.in*
http://www.iitk.ac.in/kangal/

***Abstract*: Evolutionary multi-objective optimization (EMO) methodologies have been amply applied to find a representative set of Pareto-optimal solutions in the past decade and beyond. Although there are advantages of knowing the range of each objective for Pareto-optimality and the shape of the Pareto-optimal frontier itself in a problem for an adequate decision-making, the task of choosing a single preferred Pareto-optimal solution is also an important task which has received a lukewarm attention so far. In this paper, we combine one such preference-based strategy with an EMO methodology and demonstrate how, instead of one solution, a preferred set of solutions near the reference points can be found parallelly. We propose two approaches for this task: (i) a modified EMO procedure based on the elitist non-dominated sorting GA or NSGA-II [1] and (ii) a predator-prey approach based on original grid based procedure [2]. On two-objective to 10-objective optimization test problems, the modified NSGA-II approach shows its efficacy in finding an adequate set of Pareto-optimal points. On two and three-objective problems, the predator-prey approach also demonstrate its usefulness. Such procedures will provide the decision-maker with a set of solutions near her/his preference so that a better and a more reliable decision can be made.**

Keywords: Reference point approach, interactive multi-objective method, decision-making, predator-prey approach, multi-objective optimization.

I. Introduction

For the past 15 years or so, evolutionary multi-objective optimization (EMO) methodologies have adequately demonstrated their usefulness in finding a well-converged and well-distributed set of near Pareto-optimal solutions [3, 4]. Due to these extensive studies and available source codes both commercially and freely, the EMO procedures have been popularly applied in various problem-solving tasks and have received a great deal of attention even by the classical multi-criterion optimization and decision-making communities.
However, recent studies [5] have discovered that at least one of the EMO methodologies – NSGA-II [1] – faces difficulty in solving problems with a large number of objectives. The difficulties are as follows: (i) the visualization of four or more objective space is a difficulty which may limit EMO methodologies for finding the entire Pareto-optimal set, (ii) the emphasis of *all* non-dominated solutions in a population for a large number of objectives may not produce enough selection pressure for a small-sized population to move towards the Pareto-optimal region fast enough and (iii) there is a need for a number of solutions that usually increases exponentially in the number of objectives to achieve a constant density of Pareto-optimal solutions. Although the use of a large population and a better visualization technique may extend their applications in solving five or so objectives, but if 10 or more objectives are to be solved, there exists a considerable amount of doubt to the use an EMO procedure in finding a well-representative set of Pareto-optimal solutions. In large-objective problem-solving, EMO methodologies can be put to benefit in finding a preferred and smaller set of Pareto-optimal solutions, instead of the entire frontier. This approach has a practical viewpoint and allows a decision-maker to concentrate only to those regions on the Pareto-optimal frontier which are of interest to her/him. EMO methodologies may provide an advantage over their classical counterparts for another pragmatic reason, which we discuss next.
The classical interactive multi-criterion optimization methods demand the decision-makers to suggest a reference direction or reference points or other clues [6] which result in a preferred set of solutions on the Pareto-optimal front. In these classical approaches, based on such clues, a single-objective optimization problem is usually formed and a single solution is found. A single solution (although optimal corresponding to the given clue) does not provide a good idea of the properties of solutions near the desired region of the front. By providing a clue, the decision-maker is not usually looking for a single solution, rather she/he is inter-

ested in knowing the properties of solutions which correspond to the optimum and near-optimum solutions respecting the clue. This is because while providing the clue in terms of weight vectors or reference directions or reference points, the decision-maker has simply provided a higher-level information about her/his choice. Ideally, by providing a number of such clues, the decision-maker in the beginning is interested in choosing a region of her/his interest. We here argue that instead of finding a single solution near the region of interest, if a number of solutions in the region of interest are found, the decision-maker will be able to make a better and more reliable decision. Moreover, if multiple such regions of interest can be found simultaneously, decision-makers can make a more effective and parallel search towards finding an ultimate preferred solution.

In this paper, we use the concept of reference point methodology in an EMO and attempt to find a set of preferred Pareto-optimal solutions near the regions of interest to a decision-maker. We suggest two approaches for this purpose. The modified NSGA-II approach is able to solve as many as 10 objectives effectively and the predator-prey approach with its current implementation exhibits its potential on two and three-objective optimization problems. All simulation runs on test problems and on some engineering design problems amply demonstrate their usefulness in practice and show another use of a hybrid-EMO methodology in allowing the decision-maker to solve multi-objective optimization problems better and with more confidence.

II. Preference-Based EMO Approaches

In the context of finding a preferred set of solutions, instead of the entire Pareto-optimal set, quite a few studies have been made in the past. The approach by Deb [7] was motivated by the *goal programming* idea [8] and required the DM to specify a goal or an aspiration level for each objective. Based on that information, Deb modified his NSGA approach to find a set of solutions which are closest to the supplied goal point, if the goal point is an infeasible solution and find the solution which correspond to the supplied goal objective vector, if it is a feasible one. The method did not care finding the Pareto-optimal solutions corresponding to the multi-objective optimization problem, rather attempted to find solutions satisfying the supplied goals.

The weighted-sum approach for multi-objective optimization was utilized by a number of researchers in finding a few preferred solutions. The method by Cvetkovic and Parmee [9] assigned each criterion a weight w_i, and additionally required a minimum level for dominance τ. Then, the definition of dominance was redefined as follows:

$$x \succ y \Leftrightarrow \sum_{i:f_i(x)\leq f_i(y)} w_i \geq \tau,$$

with a strict inequality for at least one objective. To facilitate specification of the required weights, they suggested a method to turn fuzzy preferences into specific quantitative weights. However, since for every criterion the dominance scheme only considers whether one solution is better than another solution, and not by how much it is better, this approach allows only a very coarse guidance and is difficult to control. Jin and Sendhoff also proposed a way to convert fuzzy preferences into weight intervals, and then used their dynamic weighted aggregation EA [10] to obtain the corresponding solutions. This approach converted the multi-objective optimization problem into a single objective optimization problem by weighted aggregation, but varied the weights dynamically during the optimization run within the relevant boundaries.

In the Guided Multi-Objective Evolutionary Algorithm (G-MOEA) proposed by Branke et al. [11], user preferences were taken into account by modifying the definition of dominance. The approach allowed the DM to specify, for each pair of objectives, maximally acceptable trade-offs. For example, in the case of two objectives, the DM could define that an improvement by one unit in objective f_2 is worth a degradation of objective f_1 by at most a_{12} units. Similarly, a gain in objective f_1 by one unit is worth at most a_{21} units of objective f_2. This information is then used to modify the dominance scheme as follows for two objectives:

$$\begin{aligned} x \succ y \quad \Leftrightarrow \quad & (f_1(x) + a_{12}f_2(x) \leq f_1(y) + a_{12}f_2(y)) \wedge \\ & (a_{21}f_1(x) + f_2(x) \leq a_{21}f_1(y) + f_2(y)), \end{aligned}$$

with inequality in at least one case. Although the idea works quite well for two objectives and was well utilized for distributed computing purposes elsewhere [12], providing all pair-wise information in a problem having a large number of objectives becomes a real difficulty.

In order to find a biased distribution anywhere on the Pareto-optimal front, a previous study [13] used a biased fitness sharing approach and implemented on NSGA. Based on a weight vector specifying the importance of one objective function over the other, a biased distribution was obtained on two-objective problems. However, the approach could not be used to obtain a biased distribution anywhere on the Pareto-optimal front and in a controlled manner.

Recently, Branke and Deb [14] suggested a modified and controllable biased sharing approach in which by specifying a reference direction (or a linear utility function), a set of Pareto-optimal solutions near the best solution of the utility function were found. To implement, all solutions were projected on to the linear hyper-plane and crowding distance values were computed by the ratio of the distances of neighboring solutions in the original objective space and on the projected hyper-plane. Thus, solutions which lie on a plane parallel to the chosen hyper-plane would have a comparatively large crowding distance and would be preferred. The complete process was shown to converge near to the optimal solution to the utility function in a number of two and three-objective optimization problems. The procedure demanded two user-defined parameters: a reference direction and a parameter which controls the extent of diversity needed in the final set of solutions.

The above preference-based procedures are useful in their own merits and are some ways to find a preferred set of Pareto-optimal solutions. However, each of the above methodologies, including the modified biased sharing approach, cannot be used for finding points corresponding to multiple preference conditions simultaneously. In this paper, we make use some of the above principles and suggest a couple of new procedures which have the following capabilities:

1. Multiple preference conditions can be specified simultaneously.
2. For each preference condition, a set of Pareto-optimal solutions is the target set of solutions, instead of one solution.
3. The method is indifferent to the shape of the Pareto-optimal frontier (such as convex or non-convex, continuous or discrete, connected or disconnected and others).
4. The method is applicable to a large number of objectives (say, 10 or more), a large number of variables, and linear or non-linear constraints.

The procedures of this paper are certain ways of finding a preferred set of solutions in an interactive multi-objective optimization problem, which are motivated by the classical reference point approach, which we discuss next.

III. Reference Point Interactive Approach

As an alternative to the value function methods, Wierzbicki [15] suggested the reference point approach in which the goal is to achieve a weakly, ϵ-properly or Pareto-optimal solution closest to a supplied reference point of aspiration level based on solving an achievement scalarizing problem. Given a reference point $\bar{\mathbf{z}}$ for an M-objective optimization problem of minimizing $(f_1(\mathbf{x}), \ldots, f_M(\mathbf{x}))$ with $\mathbf{x} \in S$, the following single-objective optimization problem is solved for this purpose:

$$\begin{array}{ll} \text{Minimize} & \max_{i=1}^{M} \left[w_i(f_i(\mathbf{x}) - \bar{z}_i) \right], \\ \text{Subject to} & \mathbf{x} \in S. \end{array} \qquad (1)$$

Here, w_i is the i-th component of a chosen weight vector used for scalarizing the objectives. Figure 1 illustrates the concept. For a chosen reference point, the closest Pareto-optimal solution (in the sense of the weighted-sum of the objectives) is the target solution to the reference point method. To make the procedure interactive and useful in practice, Wierzbicki [15] suggested a procedure in which the obtained solution $\mathbf{z}'$ is used to create M new reference points, as follows:

$$\mathbf{z}^{(j)} = \bar{\mathbf{z}} + (\mathbf{z}' - \bar{\mathbf{z}}) \cdot \mathbf{e}^{(j)}, \qquad (2)$$

where $\mathbf{e}^{(j)}$ is the j-th coordinate direction vector. For the two-objective problem shown in the figure, two such new

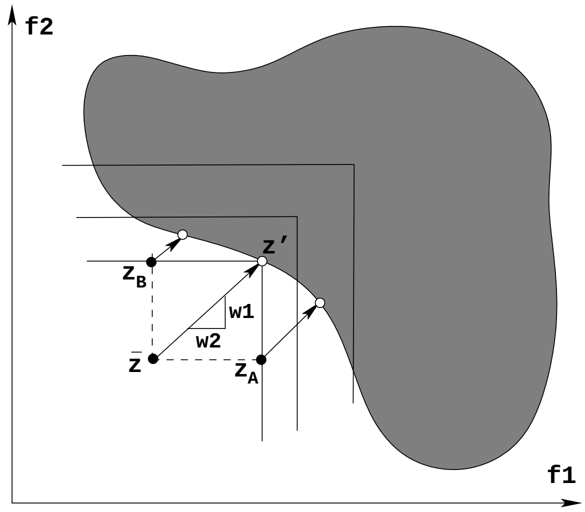

Figure. 1: Classical reference point approach.

reference points ($\mathbf{z}_A$ and $\mathbf{z}_B$) are also shown. New Pareto-optimal solutions are then found by forming new achievement scalarizing problems. If the decision-maker is not satisfied with any of these Pareto-optimal solutions, a new reference point is suggested and the above procedure is repeated. By repeating the procedure from different reference points, the decision-maker tries to evaluate the region of Pareto-optimality, instead of one particular Pareto-optimal point. It is also interesting to note that the reference point may be a feasible one (deducible from a solution vector) or an infeasible point which cannot be obtained from any solution from the feasible search space. If a reference point is feasible and is not a Pareto-optimal solution, the decision-maker may then be interested in knowing solutions which are Pareto-optimal and close to the reference point. On the other hand, if the reference point is an infeasible one, the decision-maker would be interested in finding Pareto-optimal solutions which are close to the supplied reference point.

To utilize the reference point approach in practice, the decision-maker needs to supply a reference point and a weight vector at a time. The location of the reference point causes the procedure to focus on a certain region in the Pareto-optimal frontier, whereas a supplied weight vector makes a finer trade-off among the the objectives and focuses the procedure to find a single Pareto-optimal solution (in most situations) trading-off the objectives. Thus, the reference point provides a higher-level information about the region to focus and weight vector provides a more detailed information about what point on the Pareto-optimal front to converge.

IV. Proposed Reference Point Based EMO Approach

The classical reference point approach discussed above, will find a solution depending on the chosen weight vector and is therefore subjective. Moreover, the single solution is specific to the chosen weight vector and does not provide any information about how the solution would change with a slight change in the weight vector. To find a solution for another weight vector, a new achievement scalarizing problem needs to be formed again and solved. Moreover, despite some modifications [16], the reference point approach works with only one reference point at a time. However, the decision-maker may be interested in exploring the preferred regions of Pareto-optimality for multiple reference points simultaneously.

With the above principles of reference point approaches and difficulties with the classical methods, we propose an EMO methodology by which a set of Pareto-optimal solutions near a supplied set of reference points will be found, thereby eliminating the need of any weight vector and the need of applying the method again and again. Instead of finding a single solution corresponding to a particular weight vector, the proposed procedure will attempt to a find a set of solutions in the neighborhood of the corresponding Pareto-optimal solution, so that the decision-maker can have a better idea of the region rather than a single solution.

To implement the procedure, we use the elitist non-dominated sorting GA or NSGA-II [1]. However, a similar strategy can also be adopted with any other EMO methodology. In the following, we describe an iteration of the proposed reference-point-based NSGA-II procedure (we call here as R-NSGA-II) for which the decision-maker supplies one or more reference points. As usual, both parent and offspring populations are combined together and a non-dominated sorting is performed to classify the combined population into different levels of non-domination. Solutions from the best non-domination levels are chosen front-wise as before and a modified crowding distance operator (we called here as a 'preference operator') is used to choose a subset of solutions from the last front which cannot be entirely chosen to maintain the population size of the next population. The main ideas behind choosing the preferred set of solutions are as follows:

1. Solutions closer to the reference points (in the objective space) are to be emphasized more.
2. Solutions within a ϵ-neighborhood to a near-reference-point solution are de-emphasized in order to maintain a diverse set of solutions near each reference point.

The following update to the original NSGA-II niching strategy is performed to incorporate the above two ideas:

Step 1: For each reference point, the normalized Euclidean distance (see equation (3) later) of each solution of the front is calculated and the solutions are sorted in ascending order of distance. This way, the solution closest to the reference point is assigned a rank of one.

Step 2: After such computations are performed for all reference points, the minimum of the assigned ranks is assigned as the *preference distance* to a solution. This way, solutions closest to all reference points are assigned the smallest preference distance of one. The solutions having next-to-smallest Euclidean distance to all reference points are assigned the next-to-smallest preference distance of two, and so on. Thereafter, solutions with a smaller preference distance are preferred in the tournament selection and in forming the new population from the combined population of parents and offspring.

Step 3: To control the extent of obtained solutions, an ϵ-*clearing* idea is used in the niching operator. First, a random solution is picked from the non-dominated set. Thereafter, all solutions having a sum of normalized difference in objective values of ϵ or less from the chosen solution are assigned an artificial large preference distance to discourage them to remain in the race. This way, only one solution within a ϵ-neighborhood is emphasized. Then, another solution from the non-dominated set (and is not already considered earlier) is picked and the above procedure is performed.

The above procedure provides an equal emphasis of solutions closest to each reference point, thereby allowing multiple regions of interest to be found simultaneously in a single simulation run. Moreover, the use of the ϵ-based selection strategy (which is also similar to the ϵ-dominance strategies suggested elsewhere [17, 18]) ensures a spread of solutions near the preferred Pareto-optimal regions.

In the parlance of the classical reference point approach, the above procedure is equivalent to using a weight vector emphasizing each objective function equally or using $w_i = 1/M$. If the decision-maker is interested in biasing some objectives more than others, a suitable weight vector can be used with each reference point and instead of emphasizing solutions with the shortest Euclidean distance from a reference point, solutions with a shortest weighted Euclidean distance from the reference point can be emphasized. We replace the Euclidean distance measure with the following weighted Euclidean distance measure:

$$d_{ij} = \sqrt{\sum_{i=1}^{M} w_i \left(\frac{f_i(\mathbf{x}) - \overline{z}_i}{f_i^{\max} - f_i^{\min}} \right)^2}, \qquad (3)$$

where $f_i^{\max}$ and $f_i^{\min}$ are the population maximum and minimum function values of i-th objective. Note that this weighted distance measure can also be used to find a set of preferred solutions in the case of problems having non-convex Pareto-optimal front.

V. Simulation Results

We now show simulation results on two to 10 objectives using the proposed methodology. In all simulations, we use the SBX operator with an index of 10 and polynomial mutation with an index 20 [4]. We also use a population of size 100 for all two-objective problems and run till 500 generations to investigate if a good distribution of solutions remain for a large number of iterations.

A. Two-Objective ZDT Test Problems

In this section, we consider three ZDT test problems.

1) Test Problem ZDT1

First, we consider the 30-variable ZDT1 problem. This problem has a convex Pareto-optimal front spanning continuously in $f_1 \in [0, 1]$ and follows a function relationship: $f_2 = 1 - \sqrt{f_1}$. Figure 2 shows the effect of different ϵ values on the distribution. Two reference points are chosen for this

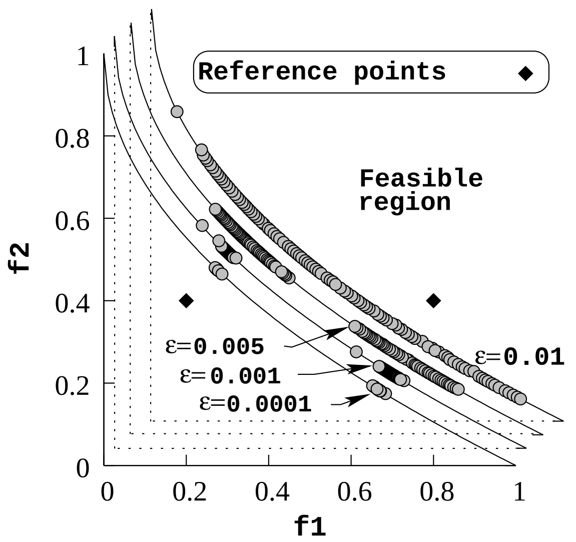

Figure. 2: Effect of ϵ in obtaining varying spread of preferred solutions on ZDT1.

problem and are shown in filled diamonds. Four different ϵ values of 0.0001, 0.001, 0.005 and 0.01 are chosen. Solutions with $\epsilon = 0.0001$ are shown on the true Pareto-optimal front. It is interesting to note how solutions close to the two chosen reference points are obtained on the Pareto-optimal front. Solutions with other ϵ values are shown with an offset to the true Pareto-optimal front. It is clear that with a large value of ϵ, the range of obtained solutions is also large. Thus, if the decision-maker would like to obtain a large neighborhood of solutions near the desired region, a large value of ϵ can be chosen. For a particular population size and a chosen number of reference points, the extent of obtained solutions gets fixed by maintaining a distance between consecutive solutions of an amount ϵ.

Next, we consider five reference points, of which two are feasible and three are infeasible. Figure 3 shows the obtained solutions with $\epsilon = 0.001$. Near all five reference points, a good extent of solutions are obtained on the Pareto-optimal front.

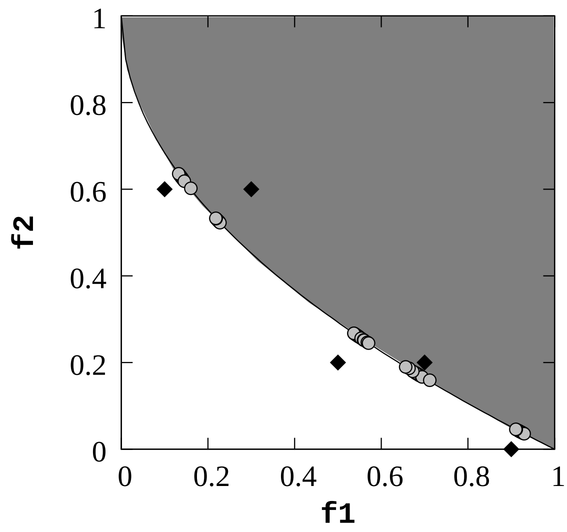

Figure. 3: Preferred solutions for five reference points with $\epsilon = 0.001$ on ZDT1.

To investigate the effect of a weight-vector in obtaining the preferred distribution (similar to the classical achievement scalarization approach), we use the normalized Euclidean distance measure given in equation 3. Figure 4 shows the obtained distribution with R-NSGA-II with $\epsilon = 0.001$ on ZDT1 problem for three different weight vectors: (0.5, 0.5), (0.2, 0.8) and (0.8, 0.2). A reference point $\bar{\mathbf{z}} = (0.3, 0.3)$ is

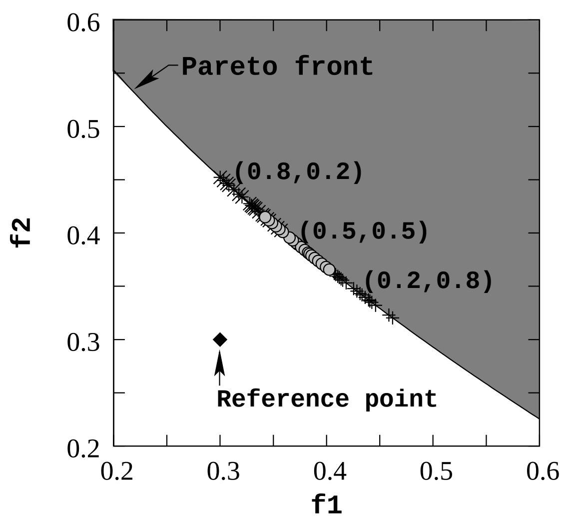

Figure. 4: Biased preferred solutions with different weight vectors around a reference point for ZDT1.

used. As expected, for the first weight vector, the obtained solutions are closest to the reference point. For the second weight vector, more emphasis on f_2 is given, thereby finding solutions which are closer to minimum of f_2. An opposite phenomenon is observed with the weight vector (0.8, 0.2), in which more emphasis on f_1 is provided. These results show that if the decision-maker is interested in biasing some objectives more than the others, a biased distribution closest to the chosen reference point can be obtained by the proposed R-NSGA-II. In all subsequent simulations, we use a uniform weight vector, however a non-uniform weight-vector can also be used, if desired.

2) Test Problem ZDT2

The 30-variable ZDT2 problem is considered next. This problem has a non-convex Pareto-optimal front ranging in $f_1.f_2 \in [0,1]$ with $f_2 = 1 - f_1^2$. Three reference points are chosen and the obtained set of points with $\epsilon = 0.001$ are shown in Figure 5. It can be clearly seen that non-convexity of the Pareto-optimal front does not cause any difficulty to the proposed methodology.

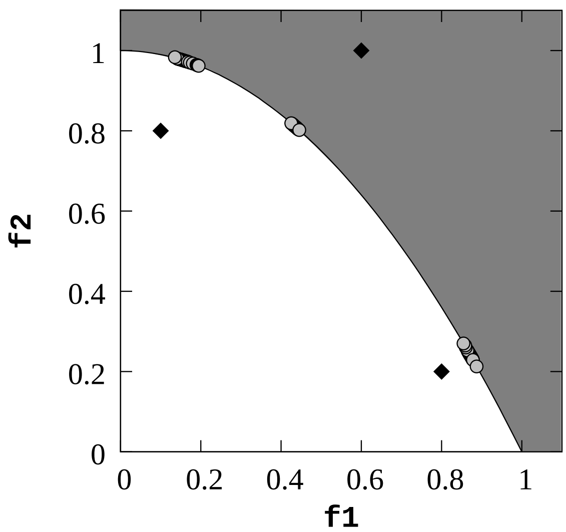

Figure. 5: Preferred solutions for three reference points with $\epsilon = 0.001$ on ZDT2.

3) Test Problem ZDT3

The 30-variable ZDT3 problem has a disconnected set of Pareto-optimal fronts. Three reference points are chosen and the obtained set of solutions found using $\epsilon = 0.001$ are shown in Figure 6. It is interesting to note that corresponding to the reference point lying between the two disconnected fronts, solutions on both fronts are discovered, providing an idea of the nature of the Pareto-optimality at the region. By using a classical approach, a solution only one solution on one of the sub-fronts would have been discovered.
This study also reveals an important matter with the proposed approach, which we discuss next. Since the complete Pareto-optimal front is not the target of the approach and since the proposed procedure emphasizes non-dominated solutions, some non-Pareto-optimal solutions can be found by

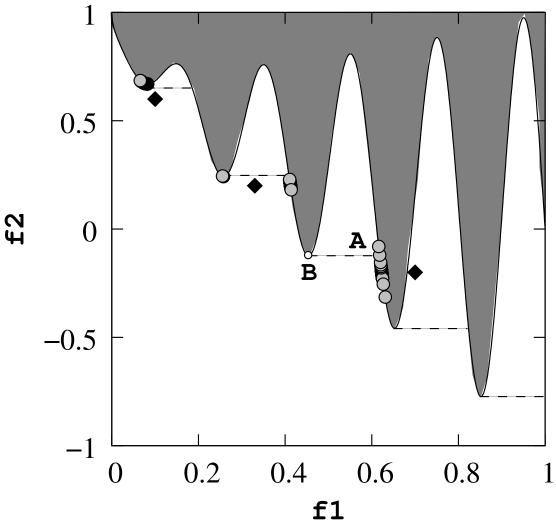

Figure. 6: Preferred solutions for three reference points with $\epsilon = 0.001$ on ZDT3.

the proposed procedure particularly in problems having non-continuous Pareto-optimal fronts. Solution A (refer Figure 6) is one such point which is not a Pareto-optimal solution but is found as a part of the final subpopulation by the proposed approach. This solution is non-dominated to the rest of the obtained solutions, but is not a member of the true Pareto-optimal set. To make this solution dominated, there exist no neighboring solution in the objective space. Only when solutions such as solution B are present in the population, such spurious solutions (like solution A) will not remain in the final population. However, the chosen reference points can be such that the solution B may not be a part of the preferred solutions. In such situations, such spurious solutions (like solution A) may appear in the final population. However, to ensure the Pareto-optimality of a solution, an ϵ-constraint approach can be applied with $f_1 \leq f_1^A$ constraint. If a solution dominating solution A is found by the ϵ-constraint approach, then solution A cannot be a member of the Pareto-optimal set. However, in this paper we realize the need of such a second-level optimization strategy for ensuring Pareto-optimality, but we do not perform such a study here.

B. Three-Objective DTLZ2 Problem

The 11-variable DTLZ2 problem has a three-dimensional, non-convex, Pareto-optimal front. We use two reference points ($(0.2, 0.2, 0.6)^T$ and $(0.8, 0.6, 1.0)^T$) as shown in Figure 7. We use $\epsilon = 0.01$ here. For this and subsequent problems, we use a population of size 500. Other parameters are set as before. A good distribution of solutions near the two reference points is obtained. This indicates the ability of the proposed procedure in solving three-objective optimization problems as well.

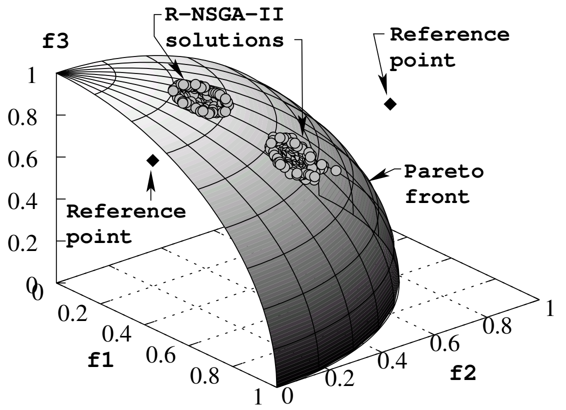

Figure. 7: Preferred solutions for two reference points with $\epsilon = 0.01$ on DTLZ2.

C. Five-Objective DTLZ2 Problem

Next, we apply the proposed procedure with $\epsilon = 0.01$ to the 14-variable DTLZ2 problem. Two reference points are chosen as follows: (i) (0.5, 0.5, 0.5, 0.5, 0.5) and (ii) (0.2, 0.2, 0.2, 0.2, 0.8). Figure 8 shows the value-path plot of the five-objective solutions. It is clear that two distinct sets of solutions near the above reference points are obtained by the proposed procedure. Since the Pareto-optimal solutions in the DTLZ2 problem satisfy $\sum_{i=1}^{M} f_i^2$ equal to one, we compute this expression for all obtained solutions and the values are found to lie within [1.000, 1.044] (at most 4.4% from one), thereby meaning that all solutions are very close to the true Pareto-optimal front.

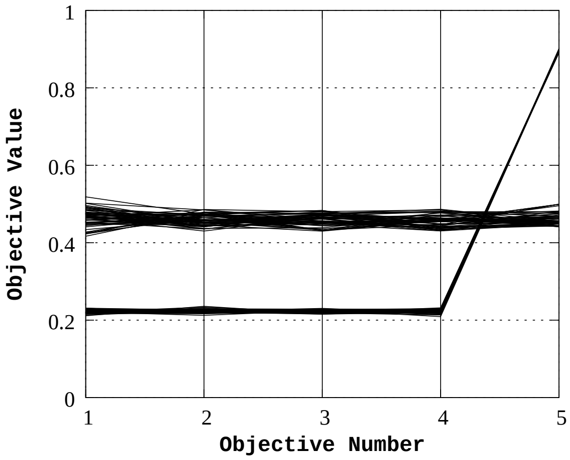

Figure. 8: Preferred solutions for two reference points with $\epsilon = 0.01$ on five-objective DTLZ2.

D. 10-Objective DTLZ2 Problem

We then attempt to solve 19-variable DTLZ2 problem with one reference point: $f_i = 0.25$ for all $i = 1, 2, \ldots, 10$. We use $\epsilon = 0.01$ and the obtained distribution is shown in Figure 9. Although the objective values can vary in [0,1], the

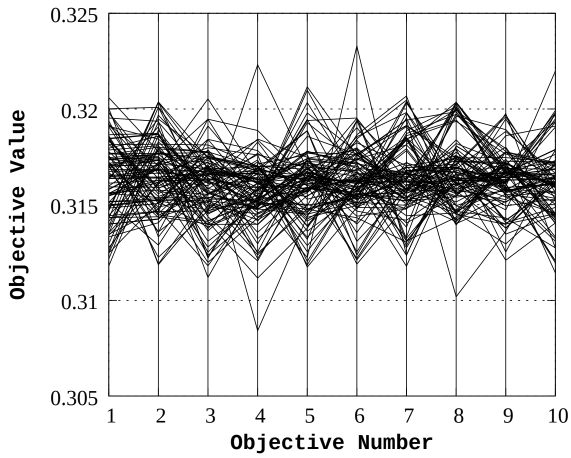

Figure. 9: Preferred solutions for one reference point with $\epsilon = 0.01$ on 10-objective DTLZ2.

points concentrates near $f_i = 1/\sqrt{10}$ or 0.316, which would be the region closest to the chosen reference point. When we compute $\sum_{i=1}^{10} f_i^2$ of all obtained solutions, they are found to be exactly equal to one, thereby meaning that all R-NSGA-II solutions are on the true Pareto-optimal front. This study shows that the proposed procedure is also able to solve a 10-objective problem, although it has been shown elsewhere [5] that the original NSGA-II faces difficulty in finding a converged and well-distributed set of solutions on the true Pareto-optimal front for the same 10-objective DTLZ2 problem. Thus, it can be concluded that if a small region on a large-dimensional Pareto-optimal front is the target, the proposed procedure is a way to find it in a reasonable amount of computations.

VI. Two Engineering Design Problems

Next, we apply the proposed methodology to two engineering design problems, each having two objectives.

A. Welded Beam Design Problem

The welded beam design problem has four real-parameter variables $\mathbf{x} = (h, \ell, t, b)$ and four non-linear constraints. One of the two objectives is to minimize the cost of fabrication and other is to minimize the end deflection of the welded beam [19]:

$$\begin{array}{rl}
\text{Minimize} & f_1(\vec{x}) = 1.10471h^2\ell + 0.04811tb(14.0+\ell),\\
\text{Minimize} & f_2(\vec{x}) = \frac{2.1952}{t^3 b},\\
\text{Subject to} & g_1(\vec{x}) \equiv 13,600 - \tau(\vec{x}) \geq 0,\\
& g_2(\vec{x}) \equiv 30,000 - \sigma(\vec{x}) \geq 0,\\
& g_3(\vec{x}) \equiv b - h \geq 0,\\
& g_4(\vec{x}) \equiv P_c(\vec{x}) - 6,000 \geq 0,\\
& 0.125 \leq h, b \leq 5.0,\\
& 0.1 \leq \ell, t \leq 10.0.
\end{array} \quad (4)$$

There are four constraints. The first constraint makes sure that the shear stress developed at the support location of the beam is smaller than the allowable shear strength of the material (13,600 psi). The second constraint makes sure that normal stress developed at the support location of the beam is smaller than the allowable yield strength of the material (30,000 psi). The third constraint makes sure that thickness of the beam is not smaller than the weld thickness from a practical standpoint. The fourth constraint makes sure that the allowable buckling load (along t direction) of the beam is more than the applied load $F = 6,000$ lbs. A violation of any of the above four constraints will make the design unacceptable. The stress and buckling terms are non-linear to design variables and are given as follows [20]:

$$\begin{aligned}
\tau(\vec{x}) &= \sqrt{(\tau')^2 + (\tau'')^2 + (\ell\tau'\tau'')/\sqrt{0.25(\ell^2+(h+t)^2)}},\\
\tau' &= \frac{6,000}{\sqrt{2}h\ell},\\
\tau'' &= \frac{6,000(14+0.5\ell)\sqrt{0.25(\ell^2+(h+t)^2)}}{2\left\{0.707h\ell(\ell^2/12+0.25(h+t)^2)\right\}},\\
\sigma(\vec{x}) &= \frac{504,000}{t^2 b},\\
P_c(\vec{x}) &= 64,746.022(1-0.0282346t)tb^3.
\end{aligned}$$

The objectives are conflicting in nature and NSGA-II is applied elsewhere to find the optimized non-dominated front to this problem [4]. Here, instead of finding the complete Pareto-optimal front, we are interested in finding the optimized trade-off regions closest to three chosen reference points:

1. (4,0.0030),
2. (20,0.0020), and
3. (40,0.0002).

Figure 10 shows the obtained solutions. To investigate where these regions are with respect to the complete trade-off front, we also show the original NSGA-II solutions with a '+'. First, the obtained preferred solutions are found to be falling on the trade-off frontier obtained using the original NSGA-II. Second, solutions close to the given reference points are found. It is interesting to note that although the second reference point is feasible. meaning that there may exist a solution vector $\mathbf{x}$, which will produce the given reference point

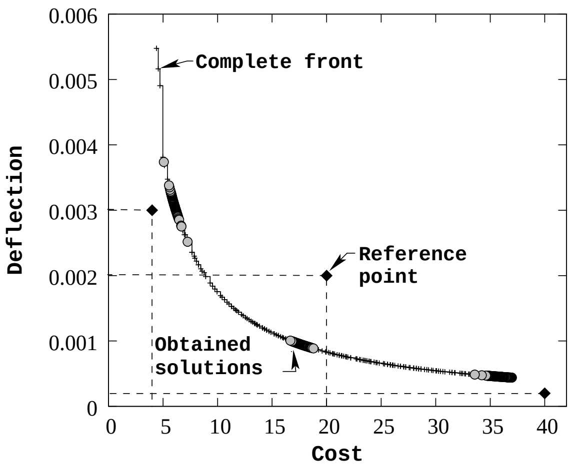

Figure. 10: Preferred solutions for three reference points with $\epsilon = 0.001$ on the welded beam design problem.

(that is, corresponding to a cost of 20 units and a deflection of 0.002 units), the task is to find, if possible, a set of solutions which are better than the given reference point in all objectives. The figure shows that the supplied reference point is not an optimal solution and there exist a number of solutions which dominate this solution $\mathbf{x}$. Although shortest distances from the reference points are preferred, the emphasis of non-dominated solutions over dominated solutions enables Pareto-optimal solutions to be found.
Thus, if the decision-maker is interested in knowing trade-off optimal solutions in three major areas (minimum cost, intermediate to cost and deflection and minimum deflection) the proposed procedure is able to find solutions near the supplied reference points, instead of finding solution on the entire Pareto-optimal front, thereby allowing the decision-maker to consider only a few solutions and that too solutions which lie in the regions of her/his interest.

B. *Spring Design Problem*

Finally, we consider another engineering design problem in which two of the three design variables are discrete in nature, thereby causing the Pareto-optimal front to have a discrete set of solutions. Diameter of the wire (d), diameter of the spring (D) and the number of turns (N) are to be found for minimizing volume of spring and minimizing the stress developed due to the application of a load. Denoting the variable vector $\mathbf{x} = (x_1, x_2, x_3) = (N, d, D)$, we write the two-objective, eight-constraint optimization problem as follows [21]:

$$\begin{array}{ll} \text{Minimize} & f_1(\vec{x}) = 0.25\pi^2 x_2{}^2 x_3 (x_1+2), \\ \text{Minimize} & f_2(\vec{x}) = \frac{8KP_{max}x_3}{\pi x_2{}^3}, \\ \text{Subject to} & g_1(\vec{x}) = l_{max} - \frac{P_{max}}{k} - 1.05(x_1+2)x_2 \geq 0, \\ & g_2(\vec{x}) = x_2 - d_{min} \geq 0, \\ & g_3(\vec{x}) = D_{max} - (x_2+x_3) \geq 0, \\ & g_4(\vec{x}) = C - 3 \geq 0, \\ & g_5(\vec{x}) = \delta_{pm} - \delta_p \geq 0, \\ & g_6(\vec{x}) = \frac{P_{max}-P}{k} - \delta_w \geq 0, \\ & g_7(\vec{x}) = S - \frac{8KP_{max}x_3}{\pi x_2{}^3} \geq 0, \\ & g_8(\vec{x}) = V_{max} - 0.25\pi^2 x_2{}^2 x_3 (x_1+2) \geq 0, \\ & x_1 \text{ is integer, } x_2 \text{ is discrete, } x_3 \text{ is continuous.} \end{array} \quad (5)$$

The parameters used are as follows:

$$\begin{array}{lll} K = \frac{4C-1}{4C-4} + \frac{0.615x_2}{x_3}, & P = 300 \text{ lb}, & D_{max} = 3 \text{ in}, \\ P_{max} = 1,000 \text{ lb}, & \delta_w = 1.25 \text{ in}, & \delta_p = \frac{P}{k}, \\ \delta_{pm} = 6 \text{ in}, & S = 189 \text{ ksi}, & d_{min} = 0.2 \text{ in}, \\ G = 11,500,000 \text{ lb/in}^2, & V_{max} = 30 \text{ in}^3, & k = \frac{Gx_2{}^4}{8x_1x_3{}^3}, \\ l_{max} = 14 \text{ in}, & C = x_3/x_2. & \end{array}$$

The 42 discrete values of d are given below:

$$\left(\begin{array}{llllll} 0.009, & 0.0095, & 0.0104, & 0.0118, & 0.0128, & 0.0132, \\ 0.014, & 0.015, & 0.0162, & 0.0173, & 0.018, & 0.020, \\ 0.023, & 0.025, & 0.028, & 0.032, & 0.035, & 0.041, \\ 0.047, & 0.054, & 0.063, & 0.072, & 0.080, & 0.092, \\ 0.105, & 0.120, & 0.135, & 0.148, & 0.162, & 0.177, \\ 0.192, & 0.207, & 0.225, & 0.244, & 0.263, & 0.283, \\ 0.307, & 0.331, & 0.362, & 0.394, & 0.4375, & 0.5. \end{array}\right)$$

The design variables d and D are treated as real-valued parameters in the NSGA-II with d taking discrete values from the above set and N is treated with a five-bit binary string, thereby coding integers in the range [1,32]. While SBX and polynomial mutation operators are used to handle d and D, a single-point crossover and bit-wise mutation are used to handle N.

We apply the R-NSGA-II with two reference points: (4, 180,000) (feasible) and (25, 20,000) (infeasible) with a uniform weight vector and with $\epsilon = 0.001$. Figure 11 shows the R-NSGA-II solutions which are found to be closer to the two reference points. The trade-off optimized solutions found by the original NSGA-II are also shown. It is interesting to note how the proposed preferred technique can be used to find a set of solutions near some chosen aspiration points, supplied by the decision-maker.

VII. A Predator-Prey Approach

Besides the above direct approach in modifying the niching operator of NSGA-II to find a preferred set of solutions, the task of finding the solutions corresponding to a set of reference points appeal a more direct natural approach to be applied. By considering the reference points as *predators* and target solutions closest to them as *preys*, we may simulate a predator-prey hunting procedure to solve the problem.

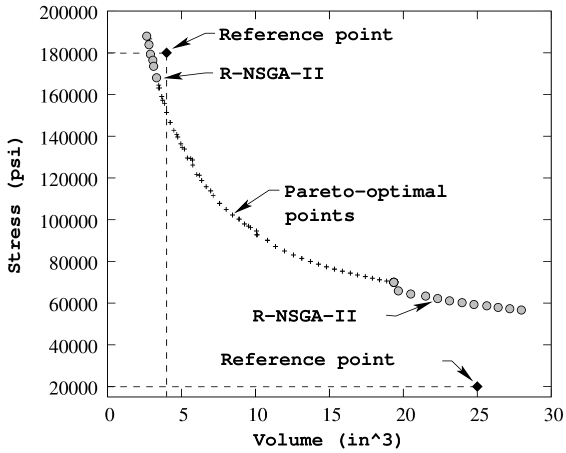

Figure. 11: Preferred solutions around two reference points for the spring design problem.

Laumanns et al. [2] suggested a predator-prey algorithm in which a predators and preys are randomly placed on a toroidal grid. Each predator works with a particular objective and deletes the worst prey it its neighborhood according to its objective function. Since every predator works with a different objective, at the end, multiple optimal solutions are expected to be present in the grid, thereby finding multiple Pareto-optimal solutions simultaneously. Later, Deb [4] extended the idea to include a weighted sum of objectives assigned to each predator. Li [22] extended the idea to introduce differing speeds of predators and preys with predators making moves more often than preys and showed an improvement in results, compared to the original method.

Here, we suggest a systematic set of modifications to the original model of Laumanns et al. [2] by introducing crossover (although the original procedure [2] suggested, but did not simulate, the use of a recombination operator), elite preservation, and an explicit diversity preservation mechanism which performed much better than the existing methodologies.

Step 1: Initialize set of preys randomly between the variable limits.

Step 2: Place these preys on the vertices of undirected connected graph.

Step 3: Place predators randomly on the vertices of the graph.

Step 4: Assign each predator with a distinct weighted sum of objectives uniformly created within $[0,1] \times [0,1] \times \ldots \times [0,1]$, so that the sum of weights is one.

Step 5: Evaluate preys around each predator and select the worst prey.

Step 6: Create two offspring by applying a crossover operation between the first and the second best preys in the neighborhood of the worst prey. Randomly choose one of the two offspring and mutate it to create the child solution.

Step 7: Child acceptance criteria:

Step 7a: If the child solution weakly dominates all existing preys, child becomes a candidate to replace the worst prey. If the child is not within the influencing region of any existing prey, it replaces the worst prey. Predator also moves to the position of the worst prey.

Step 7b: Else if the child solution is dominated by any existing prey or the child is within the influencing region of any existing prey, the child is not accepted and a new child is created by Step 6. The creation of new child and its acceptance test are continued a maximum of 10 iterations, after which the worst prey is retained. In this case, predator takes a random walk to any position in the grid.

Step 8: This completes one generation of the predator-prey algorithm. Repeat Steps 5 to 7 for the next generation.

The salient features of the proposed algorithm are as follows:

1. A weighted-sum of objectives per predator is used as a criterion for deleting the worst prey.

2. A crossover between two good solutions and a subsequent mutation are used to create a child solution.

3. The elite preservation and diversity maintenance are ensured by accepting a newly created child only when it weakly dominates all existing preys and it is not within a predefined region from existing preys.

For achieving the task for finding preferred solutions, we further modify the above procedure in the following manner:

1. Each predator is assigned to one of the reference points. Multiple predator assignment to a single reference point is also allowed and is recommended.

2. All neighboring preys are divided into two classes: (i) one which dominates the predator and (ii) the remaining solutions, as shown in Figure 12. To emphasize convergence to the Pareto-optimal front and closer to the reference points, we must deemphasize preys which are away from the Pareto-optimal front. This can be achieved by carefully comparing normalized distance of solutions from the reference points in both classes. We describe this issue next.

3. If the second set is empty, we declare the prey having the smallest normalized Euclidean distance from a predator as the worst prey. Otherwise, we find the prey in the second set having the largest normalized Euclidean distance and declare it as the worst prey. Note that in the case of a reference point residing in the infeasible region, there cannot exist any prey in class I and therefore the above procedure deletes the prey which furthest from the reference point.

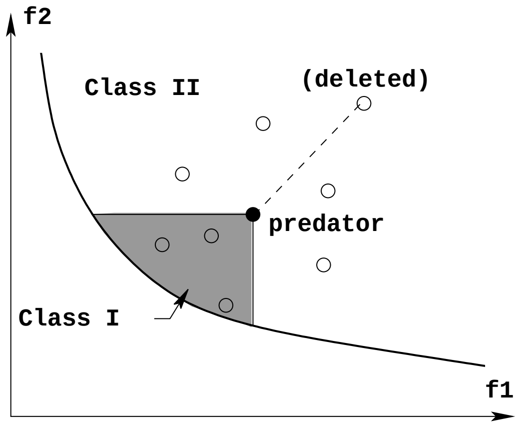

Figure. 12: Emphasizing preys near Pareto-optimal region.

4. The creation of offspring is identical to the proposed methodology. However, if only the created offspring is within a critical normalized distance (we use a value of 0.1 here) from any reference point, this offspring can be considered as a candidate for inclusion in the grid. If the offspring is not within the critical distance of any reference point, it is simply discarded.

With these modifications, we apply the procedure to a number of scenarios on the two-objective, five-variable ZDT1 test problem. All these results are taken for 300 generations. The number of preys is chosen in proportion to the number of reference points (25 times the number of reference points). In all cases, 10 predators are considered for each reference point.
Figure 13 to 16 show the final population of preys for different scenarios.
In each case, the predators (or reference points) are shown using a filled diamond. It is interesting to observe how the proposed methodology is able to find a concentrated set of Pareto-optimal solutions near each of the reference points. It is also interesting to note that the procedure works equally well for the reference point to lie inside or outside the feasible objective space.
Next, we apply the proposed predator-prey procedure to a modified three-objective DTLZ2 problem, as shown in Figure 17. To make the Pareto-optimal front a convex front (so that the weighted-sum of objectives can be assigned to each predator), we have modified the original DTLZ2 problem [23], by subtracting each function value from $(1 + g(\mathbf{x}))$. One reference point $\mathbf{z} = (0.1, 0.4, 0.7)$ is considered. 10 predators are used. A population of size 50 is used for 300 generations. The figure shows a nice concentration of 50 solutions on the true Pareto-optimal front near the supplied ref-

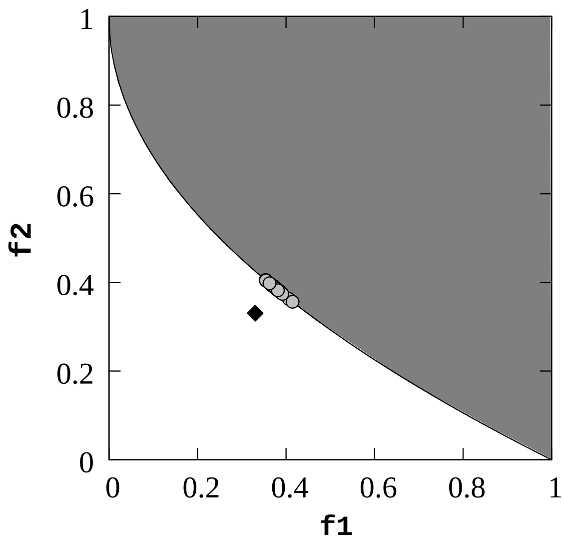

Figure. 13: Preferred solutions with an infeasible reference point.

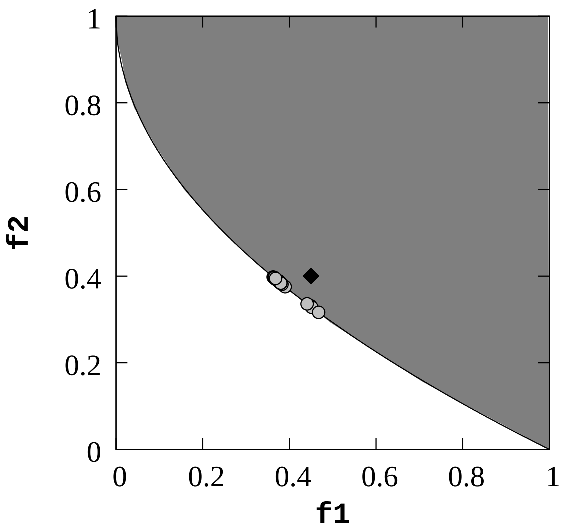

Figure. 14: Preferred solutions with a feasible reference point.

erence point. The procedure is quite fast computationally and the simulation results demonstrate its usefulness in finding a preferred set of solutions. We are currently investigating the potential of such a predator-prey procedure for handling problems with a larger number of objectives.

VIII. Extensions and Future Studies

This paper can be extended in a number of ways. In the case of R-NSGA-II, the normalization procedure in computing the Euclidean distance measure (equation 3) is an important matter. Here, we have used the population minimum and maximum objective values for this purpose. In the case of a single reference point, the EA population is likely to have a reduced diversity since all solutions are likely to concentrate in a narrow region on the Pareto-optimal set. This may cause a normalization difficulty. To avoid such a problem, in addition to emphasizing solutions near to the reference points, extreme solutions can also simultaneously emphasized (like the way they were emphasized in another study [5] to estimate the nadir objective vector). In addition to ranking population members according to the normalized distance from the reference points, they are also ranked using their distances from the extreme population members in a non-dominated front. Thereafter, the smallest of all ranks are assigned as the preference distance of a solution and solutions with smaller preference distances are emphasized as before. Since extreme solutions are also emphasized in the population, front-wise minimum and maximum objective values can now be used for computing the normalized distance measure. This way, separate subpopulations are expected to form near reference points and also near the extreme objective solutions on the Pareto-optimal front. Such a simulation will be useful in not only finding the preferred points near the reference points, but also in simultaneously getting an idea of the range of the Pareto-optimal frontier in terms of estimating the nadir point. Figure 18 shows all 100 population members after 100 generations. Other GA parameters are identical to those used before. Here, we have used a weight vector of $(1, 1)^T$ and $\epsilon = 0.001$. It can be seen that in addition to finding the preferred solutions near the singleton reference point, solutions near the extreme Pareto-optimal solutions are also found simultaneously. From these solutions, the nadir point can be computed or an idea of the range of Pareto-optimal solutions can be obtained. Similarly Figure 19 shows 100 population members after 300 generations on the three-objective DTLZ2 problem with $\epsilon = 0.01$ and an equal-component weight vector. In addition to a subpopulation near the chosen reference point, all three extreme Pareto-optimal solutions are also found by the procedure.

The ϵ-clearing strategy used in R-NSGA-II can be replaced with the grid-based ϵ-dominance principle [17] can be used. In the predator-prey approach with a reference point inside the feasible objective space, the deletion of near predator solution may eliminate a desired Pareto-optimal solution, as observed in Figure 14. This may be avoided by using rank-based selection scheme used in R-NSGA-II. Moreover, like in R-NSGA-II, a strategy controlling the extent of obtained solutions near each predator may be introduced.

IX. Conclusions

In this paper, we have addressed an important task of combining EMO methodologies with a classical multi-criterion decision-making approach to not find a single optimal solution, but to find a set of solutions near the desired region of decision-maker's interest. With a number of trade-off

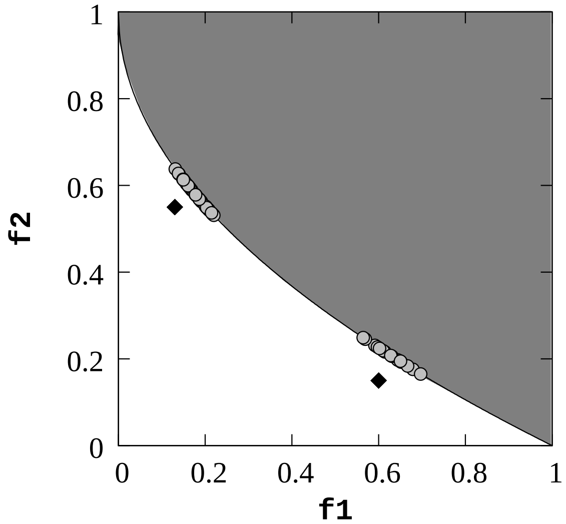

Figure. 15: Preferred solutions with two reference points.

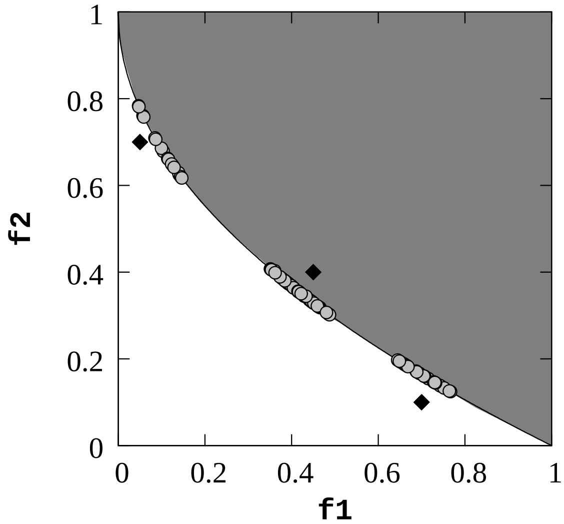

Figure. 16: Preferred solutions with three reference points.

solutions in the region of interests we have argued that the decision-maker would be able to make a better and more reliable decision than with a single solution than that with a single solution.

The reference point approach is a common methodology in multi-criterion decision-making, in which one or more reference (goal) points are specified by the decision-maker before hand. The target in such an optimization task is then to identify the Pareto-optimal region closest to the reference points. We have suggested two different approaches for this purpose. In the first approach, the niching operator of the original NSGA-II has been updated to emphasize such solutions. The proposed technique has been applied to a number of two to 10-objective optimization problems with two to five reference points and in all cases the desired set of solutions have been obtained. The approach involves a new parameter (ϵ) which controls the extent of the distribution of solutions near the closest Pareto-optimal solution.

The main crux of this paper is exploitation of the population approach of an EMO procedure in finding more than one solutions not on the entire Pareto-optimal frontier, but in the regions of Pareto-optimality which are of interest to the decision-maker. The population slots are well utilized in not only making an *implicit parallel* search [24], but also to find (i) multiple regions of interest simultaneously and (ii) multiple trade-off solutions in the close vicinity of each desired region of interest.

The second proposed approach involves another natural event of predators hunting preys of their likings. By keeping alive solutions closest to the predators (modeled for the supplied reference points) and by emphasizing non-dominated solutions for convergence to the Pareto-optimal front, the approach has been able to achieve the task quite successfully for two and three-objective optimization problems. As an immediate extension to this work, a more detailed study must be

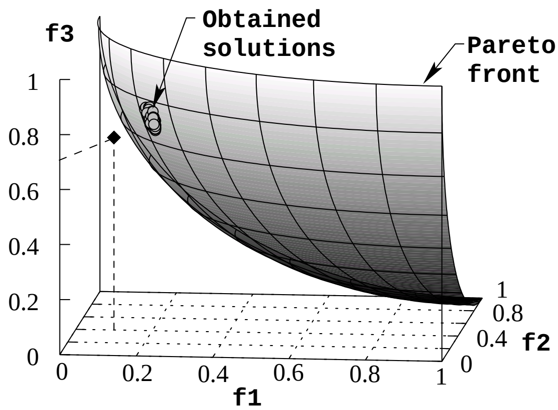

Figure. 17: Obtained solutions with the predator-prey approach for the three-objective modified DTLZ2 problem.

made to fully exploit the predator-prey approach for higher-objective problems.

Having been well demonstrated the task of finding multiple Pareto-optimal solutions in multi-objective optimization problems, the EMO researchers and applicationists should now concentrate in devising methodologies of solving the complete task of finding preferred and Pareto-optimal solutions in an interactive manner with a decision-maker. Although the ultimate target in such an activity is to come up with a single solution, the use of an EMO procedure can be well applied with a decision-making strategy in finding a set of preferred solutions in regions of interest to the decision-maker, so that the solutions in a region collectively bring out properties of the solutions there. Such an activity will then allow the decision-maker to first make a higher-level search of choosing a region of interest on the Pareto-optimal front, rather than using a single solution to focus on a particular

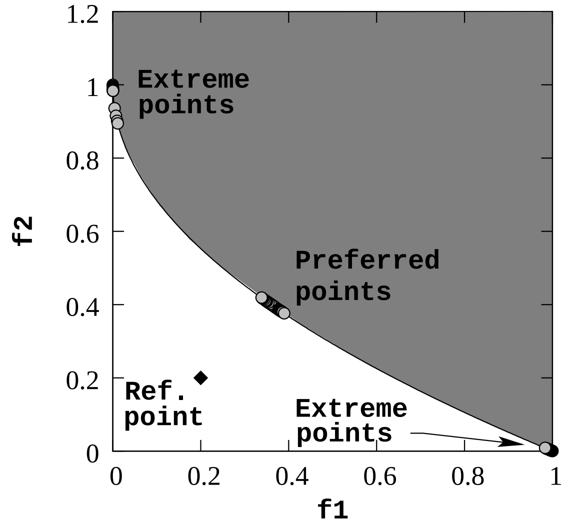

Figure. 18: Modified R-NSGA-II to find preferred as well as extreme Pareto-optimal solutions on ZDT1.

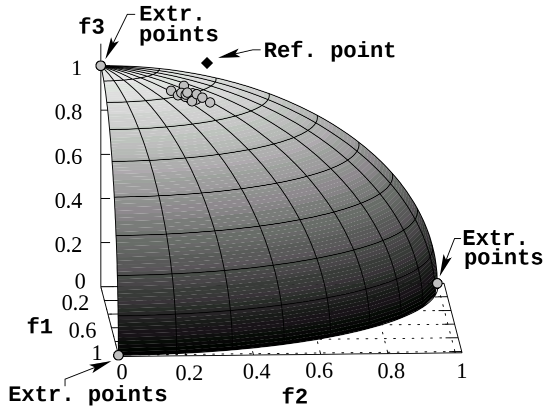

Figure. 19: Modified R-NSGA-II to find preferred as well as extreme Pareto-optimal solutions on three-objective DTLZ2 problem.

solution.

References

[1] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[2] M. Laumanns, G. Rudolph, and H. P. Schwefel. A spatial predator-prey approach to multi-objective optimization: A preliminary study. In *Proceedings of the Parallel Problem Solving from Nature, V*, pages 241–249, 1998.

[3] C. A. C. Coello, D. A. VanVeldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston, MA: Kluwer Academic Publishers, 2002.

[4] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley, 2001.

[5] K. Deb, S. Chaudhuri, and K. Miettinen. Estimating nadir objective vector quickly using evolutionary approaches. Technical Report KanGAL Report No. 2005009, Kanpur Genetic Algorithms Laboratory (KanGAL), Department of Mechanical Engineering, Indian Institute of Technology Kanpur, 2005.

[6] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer, Boston, 1999.

[7] K. Deb. Solving goal programming problems using multi-objective genetic algorithms. In *Proceedings of Congress on Evolutionary Computation*, pages 77–84, 1999.

[8] J. P. Ignizio. *Goal Programming and Extensions*. Lexington, MA: Lexington Books, 1976.

[9] D. Cvetković and I. Parmee. Evolutionary design and multi-objective optimisation. In *Proceedings of the Sixth European Congress on Intelligent Techniques and Soft Computing (EUFIT)*, pages 397–401, 1998.

[10] Y. Jin and B. Sendhoff. Fuzzy preference incorporation into evolutionary multi-objective optimization. In *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning*, volume 1, pages 26–30, 2002.

[11] J. Branke, T. Kauβler, and H. Schmeck. Guidance in evolutionary multi-objective optimization. *Advances in Engineering Software*, 32:499–507, 2001.

[12] K. Deb, P. Zope, and A. Jain. Distributed computing of pareto-optimal solutions using multi-objective evolutionary algorithms. In *Proceedings of the Second Evolutionary Multi-Criterion Optimization (EMO-03) Conference (LNCS 2632)*, pages 535–549, 2003.

[13] K. Deb. Multi-objective evolutionary algorithms: Introducing bias among pareto-optimal solutions. In A. Ghosh and S. Tsutsui, editors, *Advances in Evolutionary Computing: Theory and Applications*, pages 263–292. London: Springer-Verlag, 2003.

[14] J. Branke and K. Deb. Integrating user preferences into evolutionary multi-objective optimization. In Y. Jin, editor, *Knowledge Incorporation in Evolutionary Computation*, pages 461–477. Hiedelberg, Germany: Springer, 2004.

[15] A. P. Wierzbicki. The use of reference objectives in multiobjective optimization. In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Applications*, pages 468–486. Berlin: Springer-Verlag, 1980.

[16] A. M. J. Skulimowski. Classification and properties of dominating points in vector optimization. In P. Kleinschmidt, F. J. Radermacher, W. Scweitzer, and H. Wildermann, editors, *Methods of Operations Research 58*, pages 99–112. Frankfurt am Main, Germany: Athenum Verlag, 1989.

[17] M. Laumanns, L. Thiele, K. Deb, and Eckart Zitzler. Combining convergence and diversity in evolutionary multi-objective optimization. *Evolutionary Computation*, 10(3):263–282, 2002.

[18] K. Deb, M. Mohan, and S. Mishra. Evaluating the ϵ-domination based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions. *Evolutionary Computation Journal*, 13(4):501–525, 2005.

[19] K. Deb. An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering*, 186(2–4):311–338, 2000.

[20] G. V. Reklaitis, A. Ravindran, and K. M. Ragsdell. *Engineering Optimization Methods and Applications*. New York : Wiley, 1983.

[21] K. Deb and A. Srinivasan. Innovization: Innovation through optimization. Technical Report KanGAL Report Number 2005007, Kanpur Genetic Algorithms Laboratory, IIT Kanpur, PIN 208016, 2005.

[22] X. Li. A real-coded predator-prey genetic algorithm for multiobjective optimization. In *Proceedings of Evolutionary Multi-Criterion Optimization (EMO-2003)*, pages 207–221, 2003.

[23] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization*, pages 105–145. London: Springer-Verlag, 2005.

[24] P. Shukla and K. Deb. Comparing classical generating methods with an evolutionary multi-objective optimization method. In *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO-2005)*, pages 311–325, 2005. Lecture Notes on Computer Science 3410.