Towards an Estimation of Nadir Objective Vector Using Hybrid Evolutionary and Local Search Approaches

Kalyanmoy Deb, Kaisa Miettinen, and Shamik Chaudhuri

KanGAL Report Number 2007009

***Abstract*— Nadir objective vector is constructed with the worst Pareto-optimal objective values in a multi-objective optimization problem and is an important entity to compute because of its importance in estimating the range of objective values in the Pareto-optimal front and also in using many interactive multi-objective optimization techniques. It is needed, for example, for normalizing purposes. The task of estimating the nadir objective vector necessitates information about the complete Pareto-optimal front and is reported to be a difficult task using other approaches. In this paper, we propose certain modifications to an existing evolutionary multi-objective optimization procedure to focus its search towards the extreme objective values and combine it with a reference-point based local search approach to constitute a couple of hybrid procedures for a reliable estimation of the nadir objective vector. With up to 20-objective optimization test problems and on a three-objective engineering design optimization problem, the proposed procedures are found to be capable of finding a near nadir objective vector reliably. The study clearly shows the significance of an evolutionary computing based search procedure in assisting to solve an age-old important task of nadir objective vector estimation.**

Keywords: Nadir point, multi-objective optimization, non-dominated sorting GA, evolutionary multi-objective optimization (EMO), hybrid procedure, ideal point. Pareto optimality.

I. Introduction

In a multi-objective optimization procedure, the estimation of a nadir objective vector (or simply a nadir point) is often an important task. The nadir objective vector consists of the worst values of each objective function corresponding to the entire Pareto-optimal front. Sometimes, this point is confused with the point representing the worst objective values of the entire search space, which is often an over-estimation of the true nadir objective vector. Along with the ideal objective vector (a point constructed by the best values of each objective), the nadir objective vector is used to normalize objective functions [1], a matter often desired for an adequate functioning of multi-objective optimization algorithms in the presence of objective functions with different magnitudes. With these two extreme values, the objective functions can be scaled so that each scaled objective takes values more or less in the same range. These scaled values can be used for optimization with different algorithms like the reference point method, weighting method, compromise programming or the Tchebycheff method (see [1] and references therein). Such a scaling procedure may help in reducing the computational cost by solving the problem faster [2]. Apart from normalizing the objective function values, the nadir objective vector is also used for finding Pareto-optimal solutions in different interactive algorithms like the *guess* method [3] (where the idea is to maximize the minimum weighted deviation from the nadir objective vector), or it is otherwise an integral part of an interactive method like the NIMBUS method [1], [4]. Moreover, the knowledge of nadir and ideal objective values helps the decision-maker in adjusting her/his expectations on a realistic level by providing the range of each objective and can then be used to aid in specifying preference information in interactive methods in order to focus on a desired region. Furthermore, in visualizing Pareto-optimal front, the knowledge of the nadir objective vector is essential. Along with the ideal point, the nadir point will then provide the range of each objective in order to facilitate comparison of different Pareto-optimal solutions, that is, visualizing the trade-off information through value paths, bar charts, petal diagrams etc. [1], [5].

Researchers dealing with multiple criteria decision-making (MCDM) methodologies have suggested to approximate the nadir point using a so-called payoff table [6]. This involves computing the individual optimum solutions, constructing a payoff table by evaluating other objective values at these optimal solutions, and estimating the nadir point from the worst objective values from the table. This procedure may not guarantee a true estimation of the nadir point for more than two objectives. Moreover, the estimated nadir point can be either an over-estimation or an under-estimation of the true nadir point. For example, Iserman and Steuer [7] have demonstrated these difficulties for finding a nadir point using the payoff table method even for linear problems and emphasized the need of using a better method. Among others, Dessouky et al. [8] suggested three heuristic methods and Korhonen et al. [9] another heuristic method for this purpose. Let us point out that all these methods suggested have been developed for linear multi-objective problems where all objectives and constraints

Kalyanmoy Deb holds the Finland Distinguished Professor at Department of Business Technology, Helsinki School of Economics, PO Box 1210, FI-00101 Helsinki, Finland (Kalyanmoy.Deb@hse.fi) and Deva Raj Chair Professor at Department of Mechanical Engineering, Indian Institute of Technology Kanpur, PIN 208016, India (deb@iitk.ac.in)

Kaisa Miettinen is a Professor at Department of Business Technology, Helsinki School of Economics, P.O. Box 1210, FI-00101 Helsinki, Finland (Kaisa.Miettinen@hse.fi) and Department of Mathematical Information Technology, P.O. Box 35 (Agora), FI-40014, University of Jyväskylä, Finland (kaisa.miettinen@jyu.fi)

Shamik Chaudhuri is currently working in John F. Welch Technology Centre, Plot 122, Export Promotion Industrial Park Phase 2, Hoodi Village, Whitefield Road, Bangalore, PIN 560066, India (shamikc@gmail.com)

are linear functions of the variables.

In [10], an algorithm for deriving the nadir point is proposed based on subproblems. In other words, in order to find the nadir point for an M-objective problem, Pareto-optimal solutions of all lower-dimensional problems must first be found. Such a requirement may make the algorithm computationally impractical beyond three objectives, although Szczepanski and Wierzbicki [11] implemented the above idea using EAs and showed successful applications up to three and four objective linear optimization problems. Moreover, authors [10] did not suggest how to realize the idea in nonlinear problems. It must be emphasized that although the determination of the nadir point depends on finding the worst objective values in the set of Pareto-optimal solutions, even for linear problems, this is a difficult task [12].

Since an estimation of the nadir objective vector necessitates information about the whole Pareto-optimal front, any procedure of estimating this point should involve finding Pareto-optimal solutions. This makes the task more difficult compared to finding the ideal point [9]. Since evolutionary multi-objective optimization (EMO) algorithms can be used to find the entire or a part of the Pareto-optimal front, EMO methodologies stand as viable candidates for this task. However, some thought will reveal that an estimation of the nadir objective vector may not need finding the complete Pareto-optimal front, but only an adequate number of *extreme* Pareto-optimal solutions may be enough for this task. Based on this concept, in this paper, we suggest two modifications to an existing EMO methodology – elitist non-dominated sorting GA or NSGA-II [13] – for emphasizing to converge near to the extreme Pareto-optimal solutions, instead of emphasizing the entire the Pareto-optimal front. Thereafter, a local search methodology based on a reference point approach [14] borrowed from the multiple criteria decision-making literature is employed to enhance the convergence properties of the extreme solutions. Simulation results of this hybrid nadir point estimation procedure on problems with up to 20 objectives and on an engineering design problem amply demonstrate that one of the two approaches – the extremized crowded NSGA-II – is capable of finding a near nadir point more quickly and reliably than the other proposed method and the original NSGA-II approach of first finding the complete Pareto-optimal front and then estimating the nadir point. Results are encouraging and suggest further application of the procedure to a variety of different multiobjective optimization problems.

The rest of this paper is organized as follows. In Section II, we introduce basic concepts of multiobjective optimization and discuss the importance and difficulties of estimating the nadir point. In Section III, we describe two modified NSGA-II approaches for finding near extreme Pareto-optimal solutions. The nadir point estimation procedures proposed based on a hybrid evolutionary-cum-local-search concept is described in Section IV. Academic test problems and results are described in Section V. The use of the hybrid nadir point estimation procedure is demonstrated in Section VI by solving a test problem and an engineering design problem. Finally, the paper is concluded in Section VII.

II. Nadir Objective Vector and Difficulties of its Estimation

We consider multi-objective optimization problems involving M conflicting objectives ($f_i : \mathcal{S} \rightarrow \mathbf{R}$) as functions of decision variables $\mathbf{x}$:

$$\left.\begin{array}{ll} \text{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})\} \\ \text{subject to} & \mathbf{x} \in \mathcal{S}, \end{array}\right\} \quad (1)$$

where $\mathcal{S} \subset \mathbf{R}^n$ denotes the set of feasible solutions. A vector consisting of objective function values calculated at some point $\mathbf{x} \in \mathcal{S}$ is called an objective vector $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_M(\mathbf{x}))^T$. Problem (1) gives rise to a set of *Pareto-optimal* solutions (P^*), providing a trade-off among the objectives. In the sense of minimization of objectives, Pareto-optimal solutions can be defined as follows [1]:

Definition 1: A decision vector $\mathbf{x}^* \in \mathcal{S}$ and the corresponding objective vector $\mathbf{f}(\mathbf{x}^*)$ are Pareto-optimal if there does not exist another decision vector $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, 2, \ldots, M$ and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one index j.

Let us mention that if an objective f_j is to be maximized, it is equivalent to minimize $-f_j$. In what follows, we assume that the Pareto-optimal front is bounded. We now define a nadir objective vector as follows.

Definition 2: An objective vector $\mathbf{z}^{\text{nad}} = (z_1^{\text{nad}}, \ldots, z_M^{\text{nad}})^T$ constructed using the worst values of objective functions in the complete Pareto-optimal front P^* is called a nadir objective vector.

Hence, for minimization problems we have $z_j^{\text{nad}} = \max_{\mathbf{x} \in P^*} f_j(\mathbf{x})$. Estimation of the nadir objective vector is, in general, a difficult task. Unlike the *ideal objective vector* $\mathbf{z}^* = (z_1^*, z_2^*, \ldots, z_M^*)^T$, which can be found by minimizing each objective individually over the feasible set S (or, $z_j^* = \min_{\mathbf{x} \in \mathcal{S}} f_j(\mathbf{x})$), the nadir point cannot be formed by maximizing objectives individually over $\mathcal{S}$. To find the nadir point, Pareto-optimality of solutions used for constructing the nadir point must be first established. This makes the task of finding the nadir point a difficult one.

To illustrate this aspect, let us consider a bi-objective minimization problem shown in Figure 1. If we maximize f_1 and f_2 individually, we obtain points A and B, respectively. These two points can be used to construct the so-called *worst objective vector*. In many problems (even in bi-objective optimization problems), the nadir objective vector and the worst objective vector are not the same point, which can also be seen in Figure 1.

A. Payoff Table Method

Benayoun et al. [6] introduced the first interactive multi-objective optimization method and used a nadir point (although the authors did not use the term 'nadir'), which was to be found by using a payoff table. In this method, each objective function is first minimized individually and then a table is constructed where the i-th row of the table represents values of all other objective functions calculated at the point where the i-th objective obtained its minimum value. Thereafter, the maximum value of the j-th column can be considered as an

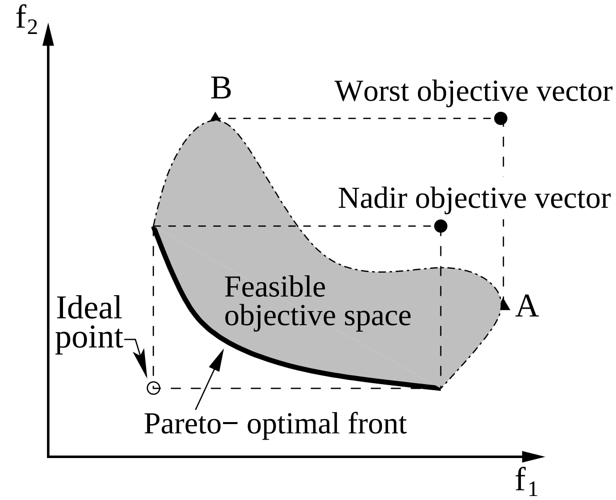

Fig. 1. The nadir and worst objective vectors.

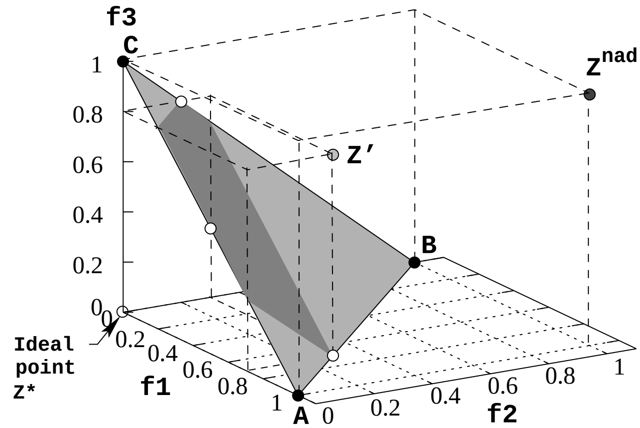

Fig. 2. Payoff table may not produce the true nadir point.

estimate of the upper bound of the j-th objective in the Pareto-optimal front and these maximum values together may be used to construct an approximation of the nadir objective vector. The main difficulty of such an approach is that solutions are not necessarily unique and thus corresponding to the minimum solution of an objective there may exist more than one solutions having different values of other objectives, in problems having more than two objectives. In these problems, the payoff table method may not result in an accurate estimation of the nadir objective vector.

Let us consider the Pareto-optimal front of a hypothetical problem involving three objective functions shown in Figure 2. The minimum value of the first objective function is zero. As can be seen from the figure, there exist a number of solutions having a value zero for function f_1 and different values of f_2 and f_3 (all solutions on the line BC). In the payoff table, when the three objectives are minimized one at a time, we may get objective vectors $\mathbf{f}^{(1)} = (0, 0, 1)^T$ (point C), $\mathbf{f}^{(2)} = (1, 0, 0)^T$ (point A), and $\mathbf{f}^{(3)} = (0, 1, 0)^T$ (point B) corresponding to minimizations of f_1, f_2, and f_3, respectively, and then the true nadir point $z^{\rm nad} = (1, 1, 1)^T$ can be found. However, if vectors $\mathbf{f}^{(1)} = (0, 0.2, 0.8)^T$, $\mathbf{f}^{(2)} = (0.5, 0, 0.5)^T$ and $\mathbf{f}^{(3)} = (0.7, 0.3, 0)^T$ (marked with open circles) are found corresponding minimizations of f_1, f_2, and f_3, respectively, a wrong estimate $\mathbf{z}' = (0.7, 0.3, 0.8)^T$ of the nadir point will be made. The figure shows how such a wrong nadir point represents only a portion (shown dark-shaded) of the Pareto-optimal front. Here we obtained an underestimation but the result may also be an overestimation of the true nadir point in some other problems.

III. Evolutionary Multi-Objective Approaches for Nadir Point Estimation

As has been discussed so far, the nadir point is associated with Pareto-optimal solutions and, thus, determining a set of Pareto-optimal solutions will facilitate the estimation of the nadir point. For the past decade or so, evolutionary multi-objective optimization (EMO) algorithms have been gaining popularity because of their ability to find multiple, wide-spread, Pareto-optimal solutions simultaneously in a single simulation run [15], [16]. Since they aim at finding a set of Pareto-optimal solutions, an EMO approach may be an ideal way to find the nadir objective vector.

Simply, a well-distributed set of Pareto-optimal solutions can be attempted to find by an EMO, as was also suggested by and an estimate of the nadir objective vector can be constructed by picking the worst values of each objective. This idea was implemented recently [11] and applied to a couple of three and four objective optimization problems. However, this *naive* procedure of first finding a representative set of Pareto-optimal solutions and then determining the nadir objective vector seems to possess some difficulties. Recall that the main purpose of the nadir objective vector, along with ideal point, is to be able to normalize different objective functions, so an interactive multi-objective optimization algorithm can be used to find the most preferred Pareto-optimal solution. Some may argue that if an EMO has already been used to find a representative Pareto-optimal set for the nadir point estimation, there is no apparent reason for constructing the nadir point for any further analysis. They can further suggest a simple approach in which the decision maker can evaluate the suitability of each obtained Pareto-optimal solution by using some higher-level information and finally choose a particular solution. However, representing and analyzing the set of Pareto optimal solutions is not trivial when we have more than two objectives in question. Furthermore, we can list several other difficulties related to the above-described simple approach. Recent studies have shown that EMO approaches using the domination principle possess a number of difficulties in solving problems having a large number of objectives [17], [18]:

1) To represent a high-dimensional Pareto-optimal front requires an exponentially large number of points [15], which, among others, increases computational cost.
2) With a large number of conflicting objectives, a large proportion of points in a random initial population are

non-dominated to each other. Since EMO algorithms emphasize *all* non-dominated solutions in a generation, a large portion of an EA population gets copied to the next generation, thereby allowing only a small number of new solutions to be included in a generation. This severely slows down the convergence of an EMO towards the true Pareto-optimal front.

3) EMO methodologies maintain a good diversity of non-dominated solutions by explicitly using a niche-preserving scheme which uses a diversity metric specifying how diverse the non-dominated solutions are. In a problem with many objectives, defining a computationally fast yet a good indicator of higher-dimensional distances among solutions becomes a difficult task. This aspect also makes the EMO approaches computationally expensive.
4) With a large number of objectives, visualization of a large-dimensional Pareto-optimal front gets difficult.

The above-mentioned shortcomings cause EMO approaches to be inadequate for finding the complete Pareto-optimal front in the first place [17]. Thus, for handling a large number of objectives, it may not be advantageous to first use an EMO approach for finding representative points of the entire Pareto-optimal front and then estimate the nadir point.

Szczepanski and Wierzbicki [11] have simulated the idea of employing multiple bi-objective optimization techniques suggested elsewhere [10] using an EMO approach to solve a three-objective test problem and construct the nadir point by accumulating all bi-objective Pareto-optimal fronts together. Although the idea seems interesting and theoretically sound, it requires $\binom{M}{2}$ bi-objective optimizations to be performed. This may be a daunting task particularly for problems having more than three or four objectives.

However, the above idea can be pushed further and instead of finding bi-objective Pareto-optimal fronts, an emphasis can be placed in an EMO approach to find only the extreme points of the Pareto-optimal front. These points are non-dominated extreme points which will be required to estimate the nadir point correctly. With this change in focus, the EMO approach can also be used to handle large-dimensional problems, particularly since the focus would be to only converge to the extreme points on the Pareto-optimal front. For the three-objective minimization problem of Figure 2, the proposed EMO approach would then distribute its population members near the extreme points A, B, and C, instead of on the entire Pareto-optimal front (on the triangle ABC), so that the nadir point can be estimated quickly. In the following subsections, we describe two EMO approaches for this purpose.

A. Worst Crowded NSGA-II Approach

In this study, we implement a couple of different nadir point estimation approaches on a particular EMO approach (NSGA-II [13]), but they can also be implemented in other state-of-the-art EMO approaches as well. Since the nadir point must be constructed from the worst objective values of Pareto-optimal solutions, it is intuitive to think of an idea in which population members having the worst objective values within a non-dominated front are emphasized. In our first approach, we employ a modified *crowing distance scheme* in NSGA-II by emphasizing the worst objective values in every non-dominated front.

In every generation, population members on every non-dominated front (having N_f members) are first sorted from minimum to maximum based on each objective (for minimization problems) and a rank equal to the position of the solution in the sorted list is assigned. In this way, a member i in a front gets a rank $R_i^{(m)}$ from the sorting in m-th objective. The solution with the minimum function value in the m-th objective gets a rank value $R_i^{(m)} = 1$ and the solution with the maximum function value in the m-th objective gets a rank value $R_i^{(m)} = N_f$. Such a rank assignment continues for all M objectives. Thus, at the end of this assignment process, each solution in the front gets M ranks, one corresponding to each objective function. Thereafter, the crowding distance d_i to a solution i in the front is assigned as the maximum of all M ranks:

$$d_i = \max\left\{R_i^{(1)}, R_i^{(2)}, \ldots, R_i^{(M)}\right\}. \quad (2)$$

The diversity preserving operator of NSGA-II emphasizes solutions having higher crowding distance value. In this way, the solution with the maximum objective value of any objective gets the best crowded distance. Like before, the NSGA-II approach emphasizes a solution if it lies on a better non-dominated front and for solutions of the same non-dominated front it emphasizes a solution with a higher crowding distance value. Thus, solutions of the final non-dominated front which could not be accepted entirely by NSGA-II's selection operator are selected based on their crowding distance value. Solutions having the worst objective value get emphasized. This dual task of selecting non-dominated solutions and solutions with worst objective values should, in principle, lead to a proper estimation of the nadir point in most problems.

However, we realize that an emphasis on the worst non-dominated solutions alone may have at least two difficulties in certain problems. First, since the focus is to find only a few solutions (instead of a complete front), the population may lose its diversity early on during the search process, thereby slowing down the progress towards the true extreme points. Moreover, if, for some reason, the convergence is a premature event to wrong solutions, the lack of diversity among population members will make it even harder for the EMO to find the necessary extreme solutions to construct the true nadir point.

The second difficulty of the worst-crowded NSGA-II approach appears in certain problems and is a more serious issue. In some problems, solutions involving the worst objective values may give rise to a correct estimate of the nadir point, but piggy-backing with one or more spurious points (which are non-optimal but non-dominated with the obtained worst points) may lead to a wrong estimate of the nadir point. We discuss this important issue with an example problem here. Consider a three-objective minimization problem shown in Figure 3, where the surface ABCD represents the Pareto-optimal front. Points A, B, C and D can be used to construct

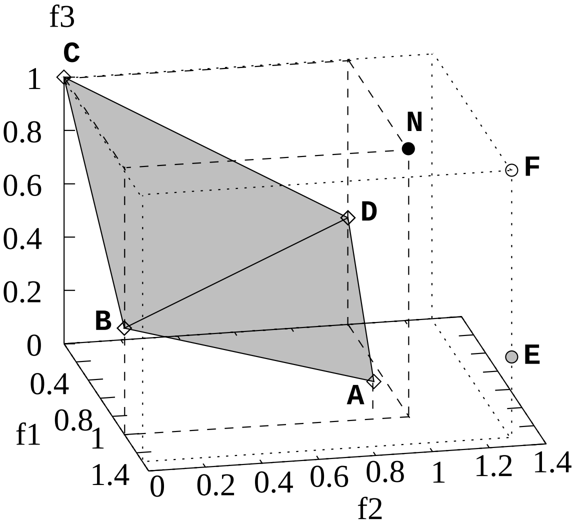

Fig. 3. A problem which may cause difficulty to the worst crowded approach.

the true nadir point $\mathbf{z}^{\rm nad} = (1,1,1)^T$. Now, by using the worst-crowded NSGA-II, we expect to find three individual worst objective values, which are points B=$(1,0,0.4)^T$ (for f_1), D=$(0,1,0.4)^T$ (for f_2) and C=$(0,0,1)^T$ (for f_3). Note that there is no motivation for the worst-crowded NSGA-II to find and maintain point A=$(0.9,0.9,0.1)^T$ in the population, as this point does not correspond to the worst value of any objective in the set of Pareto-optimal solutions. With the three points (B, C and D) in a population, a point E (with an objective vector $(1.3,1.3,0.3)^T$) if found by EA operators, will become non-dominated to points B, C, and D, and will continue to exist in the population. Thereafter, the worst-crowded NSGA-II will emphasize points C and E as extreme points and the reconstructed nadir point will become F=$(1.3,1.3,1.0)^T$, which is a wrong estimation. This difficulty could have been avoided, if the point A was included in the population.

A little thought will reveal that the point A is a Pareto-optimal solution, but corresponds to the best value of f_3. If point A is present in the population, it will dominate point E and would not allow point E to be present in the non-dominated front. Interestingly, this situation does not occur in bi-objective optimization problems. To avoid a wrong estimation of the nadir point due to the above difficulty, ideally, an emphasis on maintaining *all* Pareto-optimal solutions in the population must be made. But, since this is not practically viable, we suggest another approximate approach which is somewhat better than this worst-crowded approach.

B. Extremized Crowded NSGA-II Approach

In the extremized crowded approach proposed here, in addition to emphasizing the worst solution corresponding to each objective, we also emphasize the best solution corresponding to every objective. In this approach, solutions on a particular non-dominated front are first sorted from minimum (with rank $R_i^{(m)} = 1$) to maximum (with rank $= N_f$) based on each objective. A solution closer to either extreme objective vectors (minimum or maximum objective values) gets a higher rank compared to that of an intermediate solution. Thus, the rank of solution i for the m-th objective $R_i^{(m)}$ is reassigned as $\max\{R_i^{(m)}, N_f - R_i^{(m)} + 1\}$. Two extreme solutions for every objective get a rank equal to N_f (number of solutions in the non-dominated front), the solutions next to these extreme solutions get a rank $(N_f - 1)$, and so on. Figure 4 shows this rank-assignment procedure. After a rank is assigned to a

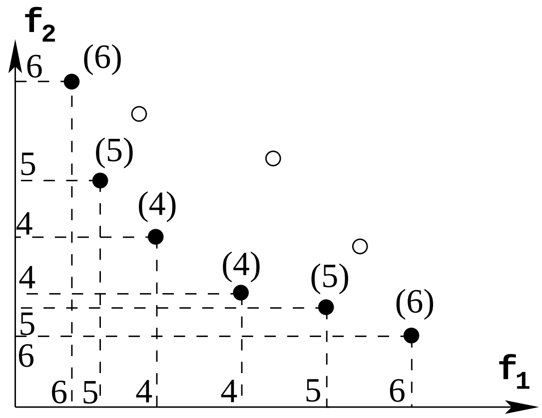

Fig. 4. Extremized crowding distance calculation.

solution by each objective, the maximum value of the assigned ranks is declared as the crowding distance, as in (2). The final crowding distance values are shown within brackets in Figure 4.

For a problem having a one-dimensional Pareto-optimal front (such as, in a bi-objective problem), the above crowding distance assignment is similar to the worst crowding distance assignment scheme (as the minimum-rank solution of one objective is the maximum-rank solution of at least one other objective). However, for problems having a higher-dimensional Pareto-optimal hyper-surface, the effect of extremized crowding is different from that of the worst crowded approach. In the three-objective problem shown in Figure 3, the extremized crowded approach will not only emphasize the extreme points A, B, C and D, but also solutions on edges CD and BC (having the smallest f_1 and f_2 values, respectively) and solutions near them. This approach has two advantages: (i) a diversity of solutions in the population may now allow genetic operators (recombination and mutation) to find better solutions and not cause a premature convergence and (ii) the presence of these extreme solutions will reduce the chance of having spurious non-Pareto-optimal solutions (like point E in Figure 3) to remain in the non-dominated front, thereby causing a more accurate computation of the nadir point. Moreover, since the intermediate portion of the Pareto-optimal front is not targeted in this approach, finding the extreme solutions should be quicker than the original NSGA-II, especially for problems having a large number of objectives and involving computationally expensive evaluation schemes.

IV. Nadir Point Estimation Procedure

The NSGA-II approach (and for this matter any other EMO method) is usually found to come closer to the Pareto-optimal front quickly and then observed to take many iterations to reach to the exact front. To enhance the performance, often NSGA-II solutions are improved by using a local search approach [15], [19]. In the context of nadir point estimation using the proposed modified NSGA-II approaches, one of the following three scenarios can occur with each obtained extreme non-dominated point:

1) It is truly an extreme Pareto-optimal solution which can contribute in estimating the nadir point accurately,
2) It gets dominated by the true extreme Pareto-optimal solution, or
3) It is non-dominated to the true extreme Pareto-optimal solution.

In the event of second and third scenario, an additional local search approach may be necessary to reach to the true extreme Pareto-optimal solution. Since, the estimation of the nadir point needs that the extreme Pareto-optimal solutions are found accurately, we suggest a hybrid, two-step procedure of first obtaining near-extreme solutions by using a modified NSGA-II procedure and then improving them by using a local search approach. The following procedure can be used with either worst-crowded or extremized-crowded NSGA-II approaches, although our initial hunch and simulation results presented later suggest the superiority of the latter approach. We call this hybrid method is our proposed nadir point estimation procedure.

Steo 1: Supply or compute ideal and worst objective vectors.

Step 2: Apply worst-crowded or extremized-crowded NSGA-II approach to find a set of non-dominated extreme points. Iterations are continued till a termination criterion (described in the next subsection), which uses ideal and worst objective vectors computed in Step 1, is met. Say, P non-dominated extreme points are found in this step.

Step 3: Apply a local search approach from each extreme point $\mathbf{x}$ to find the corresponding optimal solution $\mathbf{y}^*$ using the following augmented achievement scalarizing function:

$$\begin{array}{ll} \text{Minimize} & \max_{j=1}^{M} \bar{w}_j^{\mathbf{x}} \left(\frac{f_j(\mathbf{y}) - z_j(\mathbf{x})}{f_j^{\max} - f_j^{\min}} \right) \\ & + \rho \sum_{k=1}^{M} \bar{w}_j^{\mathbf{x}} \left(\frac{f_j(\mathbf{y}) - z_j(\mathbf{x}}{f_j^{\max} - f_j^{\min}} \right), \\ \text{subject to} & \mathbf{y} \in \mathcal{S}, \end{array} \quad (3)$$

where $f_j^{\max}$ and $f_j^{\min}$ are the minimum and maximum values of j-th objective function obtained from the set P. The j-th component of the reference point $\mathbf{z}(\mathbf{x})$ is identical to $f_j(\mathbf{x})$, except if the component corresponds to the worst value ($f_j^{\max}$) of the j-th objective. Then the following value is used: $z_j(\mathbf{x}) = f_j^{\max} + 0.5(f_j^{\max} - f_j^{\min})$. The local search is started from $\mathbf{x}$. For the extreme point $\mathbf{x}$, a pseudo-weight vector is computed as follows:

$$\bar{w}_j^{\mathbf{x}} = \max \left\{ \epsilon, \frac{(f_j(\mathbf{x}) - f_j^{\min})/(f_j^{\max} - f_j^{\min})}{\sum_{k=1}^{M} (f_k(\mathbf{x}) - f_k^{\min})/(f_k^{\max} - f_k^{\min})} \right\}. \quad (4)$$

A small value of ϵ is used to avoid a weight of zero.

Finally, construct the nadir point from the worst objective values of the extreme Pareto-optimal solutions obtained by the above procedure.

Next, we describe the details of the local search approach.

A. Reference Point Based Local Search Approach

We have suggested the use of the achievement scalarizing function derived from a reference point approach [14] as a local search approach. In this way, the reference point approach can be applied to convex or non-convex problems alike. The motivation for using the above-mentioned weight vector and the augmented achievement scalarizing function is discussed here with the help of a hypothetical bi-objective optimization problem.

Let us consider a bi-objective optimization problem shown in Figure 5, in which points A and B (close to the true extreme Pareto-optimal solutions P and Q, respectively) are found by one of the above modified NSGA-II approaches. Our

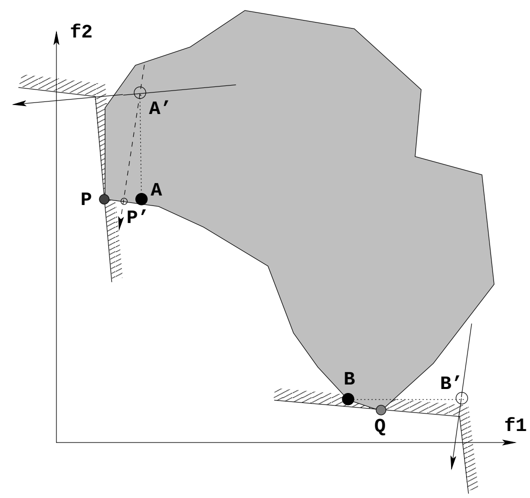

Fig. 5. The local search approach is expected to find extreme Pareto-optimal solutions exactly.

goal for employing the local search approach is to reach the corresponding true extreme point (P or Q) from each of the near-extreme points (A or B). The approach suggested above first constructs a reference point, by identifying the objective function for which the point corresponds to the worst value. In the case of point A, the worst objective vector is f_2 and for point B, it is f_1. Next, for each point, the corresponding worst function value is degraded further and the other function value is kept the same to form a reference point. For point A, the above task will construct point A′ as the reference point. Similarly, for B, the point B′ will be the reference point. The degradation of the worst objective value (suggested in Step 3 of the nadir point estimation procedure) causes the reference point to lie on the attainable part of the Pareto-optimal front which, along with the proposed weighting scheme (discussed in the next paragraph) facilitates in finding the true extreme point. If an extreme solution obtained by a modified NSGA-II approach corresponds to the worst objective value in more than one objective, all such objective values must be degraded to construct the reference point.

We can now discuss the weighting scheme suggested here. Along with the above reference point assignment scheme we suggest to choose a weight vector which will cause the achievement scalarizing problem to have its optimum solution in the corresponding extreme point. In other words, the idea of choosing an appropriate weight vector is to project the chosen reference point to the extreme Pareto-optimal solution. We have suggested equation (4) for this purpose. For point A, this approach assigns a weight vector $(\epsilon, 1)^T$. The optimization of the achievement scalarizing function can be thought as a process of moving from the reference point (A′) along a direction formed by reciprocal of weights and finding the extreme contour of the achievement scalarizing function corresponding to all feasible solutions. For point A′, this direction is marked using a solid arrow and the extreme contour is shown to correspond to the Pareto-optimal solution P. To avoid converging to a weak Pareto-optimal solution, we use the augmented achievement scalarizing function [1] with a small value of ρ here. For the near-extreme point B, the corresponding weight vector is $(1, \epsilon)^T$ and as depicted in the figure, the resulting local search solution is the other extreme Pareto-optimal solution Q. Since each of these optimizations are suggested to be started from their respective modified NSGA-II solutions, the computation of the above local search approaches is expected to be fast.

We should highlight the fact that any arbitrary weight vector may not result in finding the true extreme Pareto-optimal solution. For example, if for the reference point A′, a weight vector shown with a dashed arrow is chosen, the corresponding optimal solution of the achievement scalarizing problem will not be P, but P′. Our suggestion of the construction of reference point and corresponding weight vector together seems to be one viable way of converging to an extreme Pareto-optimal solution.

Before we leave this subsection, we discuss one further issue. It is mentioned above that the use of augmented achievement scalarizing function allows us not to converge to a weak Pareto-optimal solution by the local search approach. But, in certain problems, the approach may only allow to find an extreme *proper* Pareto-optimal solution [1] depending on the value of the parameter ρ. If for this reason any of the exact extreme points is not found, the estimated nadir point may be inaccurate. In this study, we control the accuracy of our estimated nadir point by choosing an appropriately small ρ value. If it is a problem, it is possible to solve a lexicographic achievement scalarizing function [1] instead of the local search approach described in Step 3.

B. Termination Criterion for Modified NSGA-II

Typically, an NSGA-II simulation is terminated when a pre-specified number of generations are elapsed. Here, we suggest a performance based termination criterion which causes a NSGA-II simulation to stop when the performance reaches a desirable level. The performance metric depends on a measure stating how close the estimated nadir point is to the true nadir point. However, for applying the proposed NSGA-II approaches to an arbitrary problem (for which the true Pareto-optimal front, hence the true nadir point, is not known a priori), we would need a different concept. Using the ideal point ($\mathbf{z}^*$) and the worst objective vectors ($\mathbf{z}^w$) we can define a *normalized distance* (ND) metric as follows and track the convergence property of this metric to determine the termination of a NSGA-II approach:

$$ND = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{z_i^{\text{est}} - z_i^*}{z_i^w - z_i^*}\right)^2}. \tag{5}$$

If in a problem, the worst objective vector $\mathbf{z}^w$ (refer to Figure 1) is the same as the nadir point, the normalized distance metric value will be one. For other scenarios, the normalized distance metric value will be smaller than one. Since the exact final value of this metric for finding the true nadir point is not known a priori, we record the change (Δ) in this metric value from one generation to another. When the change is not significant over a continual number of τ generations, the modified NSGA-II approach is terminated and the current non-dominated extreme solutions are sent to the next step for performing a local search.

However, in the case of solving some academic test problems, the location of the nadir objective vector is known and a simple *error* metric (E) between the estimated and the known nadir objective vectors can be used for stopping a NSGA-II simulation:

$$E = \sqrt{\sum_{i=1}^{M}\left(\frac{z_i^{\text{nad}} - z_i^{\text{est}}}{z_i^{\text{nad}} - z_i^*}\right)^2}. \tag{6}$$

To make the approach pragmatic, in this paper, we terminate a NSGA-II simulation when the error metric E becomes smaller than a predefined threshold value (η). Each non-dominated extreme solution of the final population is then sent to the local search approach for a possible improvement.

V. Results of Numerical Tests

We are now ready to describe the results of numerical tests obtained using the nadir point estimation procedure. We have chosen problems having two objectives to 20 objectives in this study. In these test problems, the entire description of the objective space and the Pareto-optimal front is known. We choose these problems to test the working of our proposed nadir point estimation procedure. Thus, in these problems, we do not perform Step 1 explicitly. Moreover, if Step 2 of the procedure (modified NSGA-II simulation) successfully finds (using the error metric ($E \leq \eta$) for determining termination of a simulation) the known nadir point, we do not employ Step 3 (local search).

In all simulations here, we compare three different approaches:

1) NSGA-II with the worst crowded approach,
2) NSGA-II with the extremized crowded approach, and
3) A naive NSGA-II approach in which first we find a set of Pareto-optimal solutions using the original NSGA-II and then estimate the nadir point from the obtained solutions.

To investigate the robustness of these approaches, parameters associated with them are kept fixed for all problems. We use the SBX recombination operator [20] with a probability of 0.9 and polynomial mutation operator [15] with a probability of $1/n$ (n is the number of variables) and a distribution index of $\eta_m = 20$. The population size and distribution index for the recombination operator (η_c) are set according to the problem and are mentioned in the respective sections. Each algorithm is run 11 times, each time starting from a different random initial population, however all proposed procedures are started with an identical set of initial populations to be fair. The number of generations required to satisfy the termination criterion ($E \leq \eta$) is noted for each simulation run and the corresponding best, median and worst number of generations are presented for a comparison. The following parameter value is used to terminate a simulation run for all test problems: $\eta = 0.01$.

A. Bi-objective Problems

As mentioned earlier, the payoff table can be reliably used to find the nadir point for a bi-objective optimization problem and there is no real need to use an evolutionary approach. However, here we still apply the nadir point estimation procedure with two modified NSGA-II approaches to three bi-objective optimization problems and compare the results with the naive NSGA-II approach mentioned above.

Three difficult bi-objective problems (ZDT test problems) described in [21] are chosen here. The ZDT3 problem is a 30-variable problem and possesses a discontinuous Pareto-optimal front. The nadir objective vector for this problem is $(0.85, 1.0)^T$. The ZDT4 problems is the most difficult among these test problems due to the presence of 99 local non-dominated fronts, which an algorithm must overcome before reaching the global Pareto-optimal front. This is a 10-variable problem with the nadir objective vector located at $(1, 1)^T$. The test problem ZDT6 is also a 10-variable problem with a non-convex Pareto-optimal front. This problem causes a non-uniformity in the distribution of solutions along the Pareto-optimal front. The nadir objective vector of this problem is $(1, 0.92)^T$. In all these problems, the ideal point ($\mathbf{z}^*$) corresponds to a function value of zero for each objective.

Table I shows the number of generations needed to find a near nadir point (within $\eta = 0.01$) by different approaches. For ZDT3, we use a recombination index of $\eta_c = 2$ and for ZDT4 and ZDT6, we use $\eta_c = 10$. It is clear from the results that the performance indicators of the worst crowded and extremized crowded NSGA-II are more or less the same and are slightly better than those of the naive NSGA-II approach in more complex problems (ZDT4 and ZDT6).

Figures 6 and 7 present how the error metric value reduces with the generation counter for problems ZDT4 and ZDT6, respectively. All the three approaches are compared in these figures with an identical termination criterion. It is observed that the convergence patterns are almost the same for all of them. Based on these results, we can conclude that for bi-objective test problems used in this study, the extremized crowded, the worst crowded, and the naive NSGA-II approaches perform quite equally. As mentioned before, there is no real need of using an evolutionary algorithm procedure, because the payoff table method works well in such bi-objective problems.

B. Problems with More Objectives

To test Step 2 of the nadir point estimation procedure on three and more objectives, we choose three DTLZ test problems [22]. These problems are designed in a manner so that they can be extended to any number of objectives. The first problem, DTLZ1, is constructed to have a linear Pareto-optimal front. The true nadir objective vector is $\mathbf{z}^{\rm nad} = (0.5, 0.5, \ldots, 0.5)^T$ and the ideal objective vector is $\mathbf{z}^* = (0, 0, \ldots, 0)^T$. The Pareto-optimal front of the second test problem, DTLZ2, is a quadrant of a unit sphere centered at the origin of the objective space. The nadir objective vector is $\mathbf{z}^{\rm nad} = (1, 1, \ldots, 1)^T$ and the ideal objective vector is $\mathbf{z}^* = (0, 0, \ldots, 0)^T$. The third test problem, DTLZ5, is somewhat modified from the original DTLZ5 and has a one-dimensional Pareto-optimal curve in the M-dimensional space [17]. The ideal objective vector is at $\mathbf{z}^* = (0, 0, \ldots, 0)^T$ and the nadir objective vector is at $\mathbf{z}^{\rm nad} = \left((\frac{1}{\sqrt{2}})^{M-2}, (\frac{1}{\sqrt{2}})^{M-2}, (\frac{1}{\sqrt{2}})^{M-3},\ (\frac{1}{\sqrt{2}})^{M-4}, \ldots, (\frac{1}{\sqrt{2}})^{0}\right)^T$.

1) Three-Objective DTLZ Problems: All three approaches are run with 100 population members for problems DTLZ1, DTLZ2 and DTLZ5 involving three objectives. Table II shows the number of generations needed to find a solution close (within a error metric value of $\eta = 0.01$ or smaller) to the true nadir point. It can be observed that the worst crowded NSGA-II and the extremized crowded NSGA-II perform in a more or less similar way when compared to each other and are somewhat better than the naive NSGA-II. In the DTLZ5 problem, despite having three objectives, the Pareto-optimal front is one-dimensional. Thus, the naive NSGA-II approach performs as well as the proposed approaches.

To show the difference between the working principles of the modified NSGA-II approaches and the naive NSGA-II approach, we show the final populations for the extremized crowded NSGA-II and the naive NSGA-II for DTLZ1 and DTLZ2 in Figures 8 and 9, respectively. Similar results can be found for the worst crowded NSGA-II approach, but are not shown here for brevity. It is clear that the extremized crowded NSGA-II concentrates its population members near the extreme regions of the Pareto-optimal front, so that a quicker estimation of the nadir point is possible to achieve. However, in the case of the naive NSGA-II approach, a distributed set of Pareto-optimal solutions is first found using the original NSGA-II (as shown in the figure) and the nadir point is constructed from these points. Since the intermediate points do not help in constructing the nadir objective vector, the naive NSGA-II approach is expected to be slow, particularly for problems having a large number of objectives.

2) Five-Objective DTLZ Problems: Next, we study the performance of all three NSGA-II approaches on DTLZ problems involving five objectives. In Table III, we collect information about results as in previous subsections. It is now quite evident from Table III that the modifications proposed to the NSGA-II approach perform much better than the naive NSGA-II

TABLE I

COMPARATIVE RESULTS FOR BI-OBJECTIVE PROBLEMS.

Test Problem	Pop. size	Number of generations								
		NSGA-II			Worst crowd. NSGA-II			Extr. crowd. NSGA-II		
		Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
ZDT3	100	33	40	55	33	40	113	28	36	45
ZDT4	100	176	197	257	148	201	224	165	191	219
ZDT6	100	137	151	161	125	130	143	126	132	135

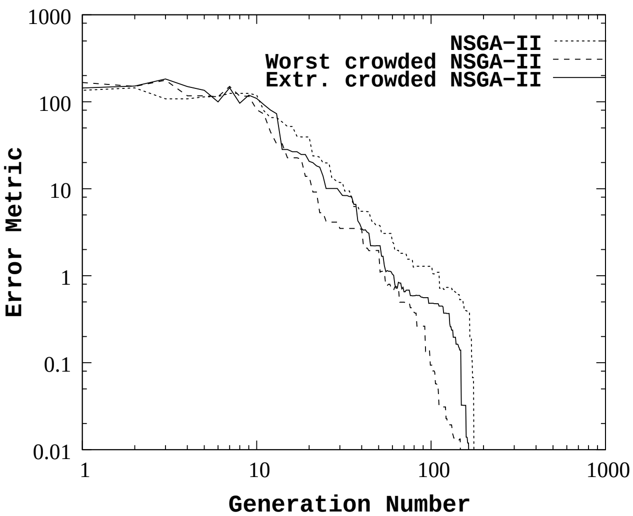

Fig. 6. The error metric for ZDT4.

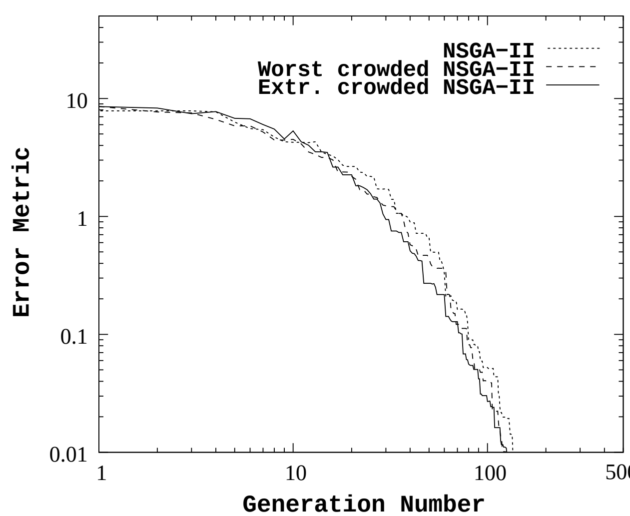

Fig. 7. The error metric for ZDT6.

TABLE II

COMPARATIVE RESULTS FOR DTLZ PROBLEMS WITH THREE OBJECTIVES.

Test problem	Pop. size	Number of generations								
		NSGA-II			Worst crowd. NSGA-II			Extr. crowd. NSGA-II		
		Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
DTLZ1	100	223	366	610	171	282	345	188	265	457
DTLZ2	100	75	111	151	38	47	54	41	49	55
DTLZ5	100	63	80	104	59	74	86	62	73	88

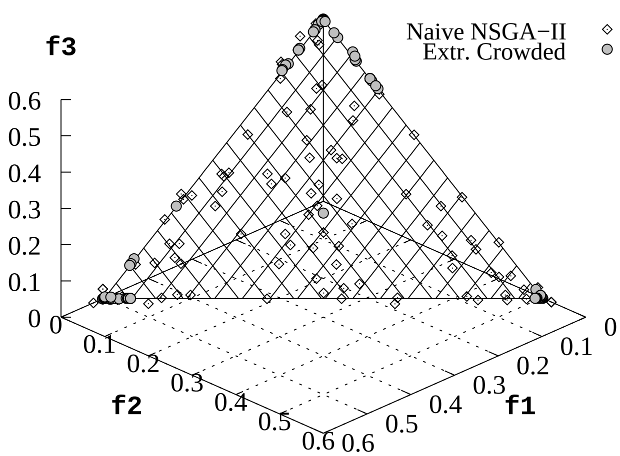

Fig. 8. Populations obtained using extremized crowded and naive NSGA-II for DTLZ1.

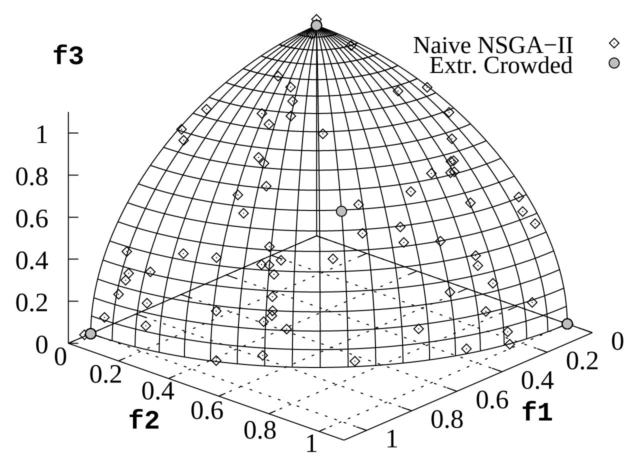

Fig. 9. Populations obtained using extremized crowded and naive NSGA-II for DTLZ2.

TABLE III

COMPARATIVE RESULTS FOR FIVE-OBJECTIVE DTLZ PROBLEMS.

Test problem	Pop. size	Number of generations								
		NSGA-II			Worst crowd. NSGA-II			Extr. crowded NSGA-II		
		Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
DTLZ1	100	2,342	3,136	3,714	611	790	1,027	353	584	1,071
DTLZ2	100	650	2,142	5,937	139	166	185	94	114	142
DTLZ5	100	52	66	77	51	66	76	49	61	73

approach. For example, for the DTLZ1 problem, the best simulation of NSGA-II takes 2,342 generations to estimate the nadir point, whereas the extremized crowded NSGA-II requires only 353 generations and the worst-crowded NSGA-II 611 generations. In the case of the DTLZ2 problem, the trend is similar. The median generation counts of the modified NSGA-II approaches for 11 independent runs are also much better than those of the naive NSGA-II approach.

The difference between the worst crowded and extremized crowded NSGA-II approaches is also clear from the table. For a problem having a large number of objectives, the extremized crowded NSGA-II emphasizes both best and worst extreme solutions for each objective maintaining an adequate diversity among the population members. The NSGA-II operators are able to exploit such a diverse population and make a faster progress towards the extreme Pareto-optimal solutions needed to estimate the nadir point correctly. However, on the DTLZ5 problem, the performance of all three approaches is similar due to the one-dimensional nature of the Pareto-optimal front. Figures 10 and 11 show the convergence of the error metric value for the best runs of the three algorithms on DTLZ1 and DTLZ2, respectively. The superiority of the extremized crowded NSGA-II approach is clear from these figures. These results imply that for a problem having more than three objectives, an emphasis on the extreme Pareto-optimal solutions (instead of all Pareto-optimal solutions) is a faster approach for locating the nadir point.

So far, we have demonstrated the ability of the nadir point estimation procedure in converging close to the nadir point by tracking the error metric value which requires the knowledge of the true nadir point. It is clear that this metric cannot be used in an arbitrary problem. We have suggested a normalized distance metric for this purpose. To demonstrate how the normalized distance metric can be used as a termination criterion, we record this metric value at every generation for both extremized crowded NSGA-II and the naive NSGA-II simulations and plot them in Figures 12 and 13 for DTLZ1 and DTLZ2, respectively. Similar trends were observed for the worst crowded NSGA-II, but for clarity the results are not superimposed in the figures here. To show the variation of the metric value over different initial populations, the region between the best and the worst normalized distance metric values is shaded and the median value is shown with a line. Recall that this metric requires the worst objective vector. For the DTLZ1 problem, the worst objective vector is computed to be $z_i^w = 551.45$ for all each objective i. Figure 12 shows that the normalized distance (ND) metric value converges to 0.00091. When we compute the normalized distance metric value by substituting the estimated nadir objective vector with the true nadir objective vector in equation (5), an identical value of ND is computed. Similarly, for DTLZ2, the worst objective vector is found to be $z_i^w = 3.25$ for $i = 1, \ldots, 5$. Figure 13 shows that the normalized distance metric (ND) value converges to 0.286, which is identical to that computed by substituting the estimated nadir objective vector with the true nadir objective vector in equation (5). Thus, we can conclude that in both problems, the convergence of the extremized crowded NSGA-II is on the true nadir point.

The rate of convergence of both approaches is also interesting to note from Figures 12 and 13. In both problems, the extremized crowded NSGA-II converges to the true nadir point quicker than the naive NSGA-II.

3) Ten-Objective DTLZ Problems: Next, we consider the three DTLZ problems for 10 objectives. Table IV presents the numbers of generations required to find a point close (within $\eta = 0.01$) to the nadir point by the three approaches for DTLZ problems with ten objectives. It is clear that the extremized crowded NSGA-II approach performs an order of magnitude better than the naive NSGA-II approach and is also better than the worst crowded NSGA-II approach. Both the DTLZ1 and DTLZ2 problems have 10-dimensional Pareto-optimal fronts and the extremized crowded NSGA-II makes a good balance of maintaining diversity and emphasizing extreme Pareto-optimal solutions so that the nadir point estimation is quick. In the case of the DTLZ2 problem with ten objectives, the naive NSGA-II could not find the nadir objective vector even after 50,000 generations (and achieved an error metric value of 5.936). Figure 14 shows a typical convergence pattern of the extremized crowded NSGA-II and the naive NSGA-II approaches on the 10-objective DTLZ1. The figure demonstrates that for a large number of generations the estimated nadir point is away from the true nadir point, but after some generations (around 1,000 in this problem) the estimated nadir point comes quickly near the true nadir point. To understand the dynamics of the movement of the population in the best performed approach (the extremized crowded NSGA-II) with the generation counter, we count the number of solutions in the population which dominate the true nadir point and plot this quantity in Figure 14. In DTLZ1, it is seen that the first point dominating the true nadir point appears in the population at around 750 generations. Thereafter, when an adequate number of such solutions appear in the population, the population very quickly converges near the extreme Pareto-optimal front for correctly estimating the nadir point. There is another matter which also helps the extremized crowded NSGA-II to converge quickly to the desired extreme points. Since extreme solutions

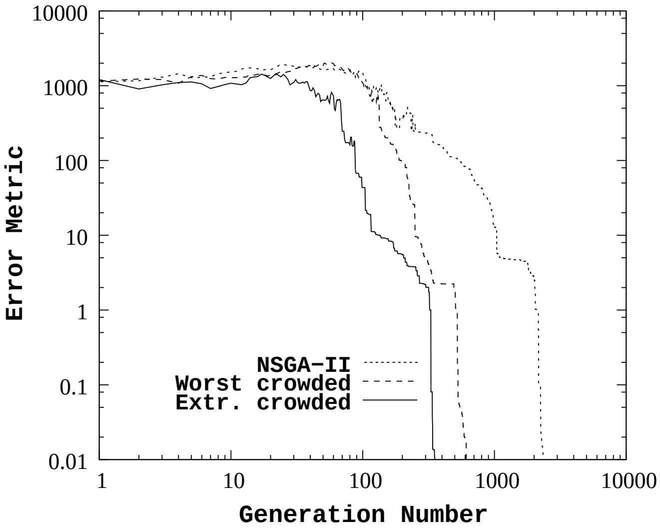

Fig. 10. The error metric on five-objective DTLZ1.

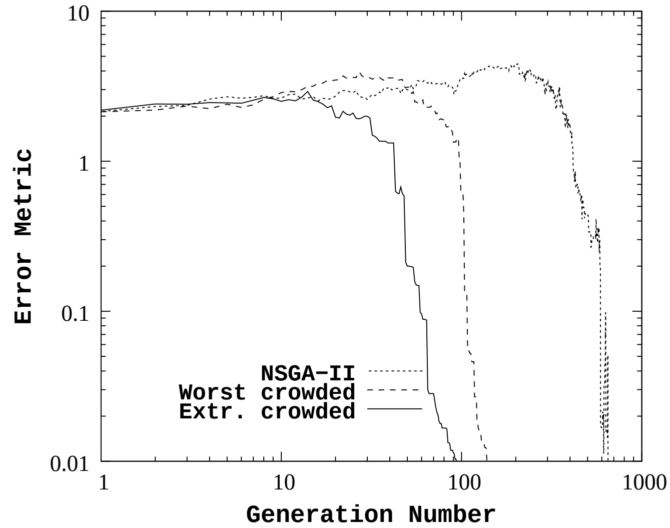

Fig. 11. The error metric on five-objective DTLZ2.

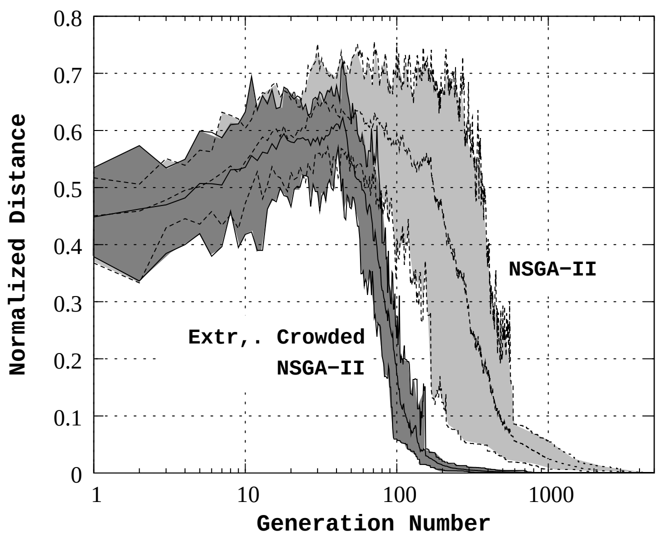

Fig. 12. Normalized distance metric of two methods on five-objective DTLZ1.

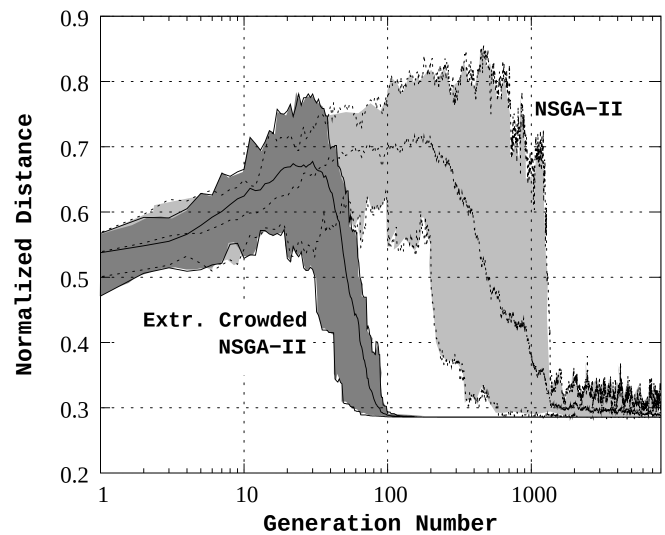

Fig. 13. Normalized distance metric of two methods on five-objective DTLZ2.

TABLE IV

COMPARATIVE RESULTS FOR 10-OBJECTIVE DTLZ PROBLEMS.

Test problem	Pop size	Number of generations								
		NSGA-II			Worst crowd. NSGA-II			Extr. crowd. NSGA-II		
		Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
DTLZ1	200	17,581	21,484	33,977	1,403	1,760	2,540	1,199	1,371	1,790
DTLZ2	200	–	–	–	520	823	1,456	388	464	640
DTLZ5	200	45	53	60	43	53	57	45	51	64

are forced to survive in the population by the ranking selection scheme, a *niching* phenomenon occurs in which multiple local niches near the extreme solutions are formed and maintained. The crossover and mutation operators acting on these niches independently then help focus on these regions more closely than in the naive NSGA-II approach and eventually cause to converge close to the true extreme Pareto-optimal solutions quickly. A similar phenomenon occurs for the worst crowded NSGA-II, but is not plotted in the same figure for clarity.

C. Scale-up Performance

In this subsection, we investigate the overall function evaluations required to reach near the true nadir point on DTLZ1 and DTLZ2 test problems having three to 20 objectives. As before, we restrict the error metric (E) value to reach below

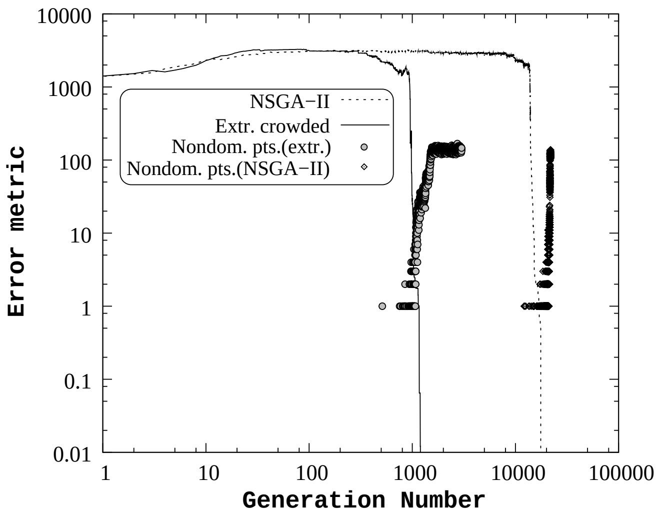

Fig. 14. Performance of three methods on 10-objective DTLZ1.

a threshold of 0.01 to determine termination of a procedure. Here, we investigate the scale-up performance of the extremized crowded NSGA-II alone and compare it against that of the naive NSGA-II approach. Since the worst crowded NSGA-II did not perform well on 10-objective DTLZ problems compared to the extremized crowded NSGA-II approach, we do not apply it here.

Figure 15 plots the best, median, and worst of 11 runs of the extremized crowded NSGA-II and the naive NSGA-II on DTLZ1. First of all, the figure clearly shows that the naive NSGA-II is unable to scale up to 15 or 20 objectives. In the case of 15-objective DTLZ1, the naive NSGA-II's performance is more than two orders of magnitude worse than that of the extremized crowded NSGA-II. For this problem, the naive NSGA-II with more than 200 million function evaluations obtained a front having a poor error metric value of 12.871 from the true nadir point. Due to the poor performance of the naive NSGA-II approach on the 15-objective problem, we did not apply it to the 20-objective DTLZ1 problem.

Figure 16 shows the performances on DTLZ2. After 670 million function evaluations, the naive NSGA-II was still not able to come close (with an error metric value of 0.01) to the true nadir point on the 10-objective DTLZ2 problem. However, the extremized crowded NSGA-II took an average of 99,000 evaluations to achieve the task. Because of the computational inefficiencies associated with the naive NSGA-II approach, we did not perform any simulation for 15 or more objectives, but the extremized crowded NSGA-II could find the nadir point up to the 20-objective DTLZ2 problem.

The nature of the plots for the extremized crowded NSGA-II in both problems is found to be sub-linear on logarithmic axes. This indicates a lower than exponential scaling property of the proposed extremized crowded NSGA-II. It is important to emphasize here that estimating the nadir point requires identification of the worst Pareto-optimal solutions. Since this requires that an evolutionary approach essentially puts its population members on the Pareto-optimal front, an adequate computational effort must be spent to achieve this task. However, results shown earlier for two to 10-objective problems have indicated that the computational effort needed by the extremized crowded NSGA-II approach is smaller when compared to the naive NSGA-II. It is worth pointing out here that decision makers do not necessarily want to or are not necessarily able to consider problems with very many objectives. However, the results of this study show a clear difference even with smaller problems involving, for example, five objectives.

VI. Results: Extremized Crowded NSGA-II with Local Search

Now, we apply the complete nadir point estimation procedure which makes a serial application of a modified NSGA-II approach followed by the local search approach on two problems. The first problem is a numerical test problem taken from the literature and the second problem is an important problem involving the design of a welded beam.

A. Problem KM

We consider a three-objective optimization problem, which provides difficulty for the payoff table method to estimate the nadir point. This problem was used in another study [23]:

$$\begin{array}{ll} \text{Minimize} & \left\{ \begin{array}{l} -x_1 - x_2 + 5 \\ \frac{1}{5}(x_1^2 - 10x_1 + x_2^2 - 4x_2 + 11) \\ (5 - x_1)(x_2 - 11) \end{array} \right\}, \\ \text{subject to} & 3x_1 + x_2 - 12 \le 0, \\ & 2x_1 + x_2 - 9 \le 0, \\ & x_1 + 2x_2 - 12 \le 0, \\ & 0 \le x_1 \le 4, \quad 0 \le x_2 \le 6. \end{array} \tag{7}$$

Individual minimizations of objectives reveal the following three objective vectors: $(-2, 0, -18)^T$, $(0, -3.1, -14.25)^T$ and $(5, 2.2, -55)^T$, thereby identifying the vector $\mathbf{z}^* = (-2, -3.1, -55)^T$ as the ideal objective vector. The payoff table method will find $(5, 2.2, -14.25)^T$ as the estimated nadir point from these minimization results. Another study [24] used a grid-search strategy (computationally possible due to the presence of only three objectives) of creating a number of feasible solutions systematically and construct the nadir point from the solutions obtained. The estimated nadir point was $(5, 4.6, -14.25)^T$ for this problem, which is different from that obtained by the payoff table method. We now employ our nadir point estimation procedure to find the nadir point for this problem.

Step 1 of the procedure finds $\mathbf{z}^* = (-2, -3.1, -55)^T$ and $\mathbf{z}^w = (5, 4.6, -14.25)^T$. In Step 2 of the procedure, we employ the extremized crowded NSGA-II and find four non-dominated extreme solutions, as shown in the first column of Table V. It is interesting to note that the fourth solution is not needed to estimate the nadir point, but the extremized principle keeps this extreme solution corresponding to f_1 to possibly eliminate spurious solutions which may otherwise stay in the population and provide a wrong estimate of the nadir point (See Figure 3 for a discussion).

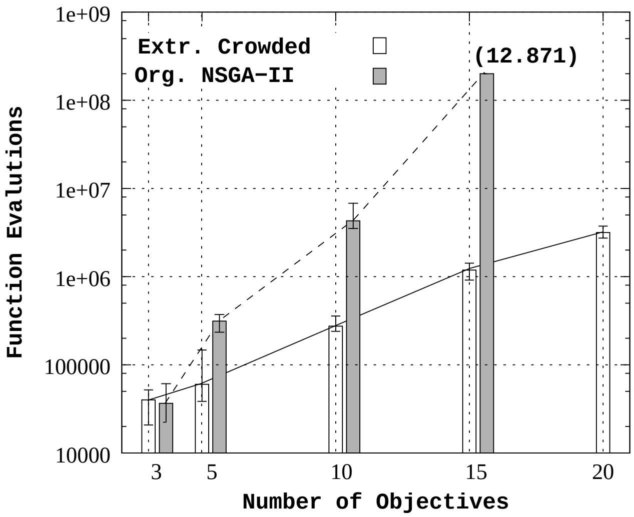

Fig. 15. Function evaluations versus number of objectives for DTLZ1.

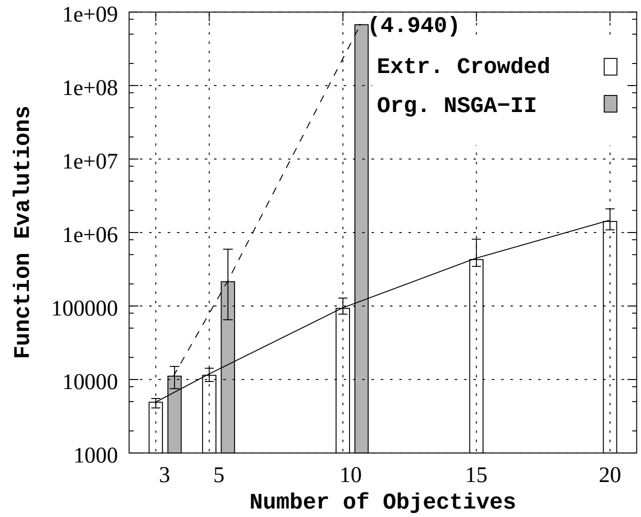

Fig. 16. Function evaluations versus number of objectives for DTLZ2.

TABLE V

EXTREMIZED CROWDED NSGA-II AND LOCAL SEARCH METHOD ON PROBLEM KM.

	$\mathbf{x}$	Objective vector	$\mathbf{w}$	$\mathbf{z}$	Extreme point
1	$(0,0)^T$	$(5, 2.2, -55)^T$	$(1, 0.688, 0.001)^T$	$(8.5, 2.2, -55)^T$	$(5, 2.2, -55)^T$
2	$(3.508, 1.477)^T$	$(0.015, -3.1, -14.212)^T$	$(0.288, 0.001, 1)^T$	$(0.015, -3.1, 6.182)^T$	$(0, -3.1, -14.25)^T$
3	$(0,6)^T$	$(-1, 4.6, -25)^T$	$(0.143, 1, 0.736)^T$	$(-1, 8.450, -25)^T$	$(-1, 4.6, -25)^T$
4	$(2,5)^T$	$(-2, 0, -18)^T$			

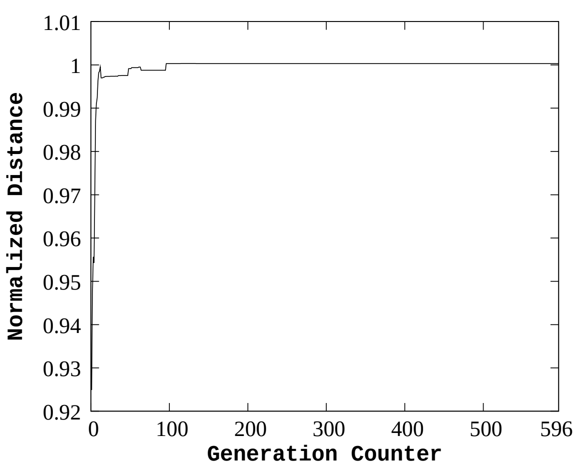

Fig. 17. Normalized distance metric with generation for problem KM.

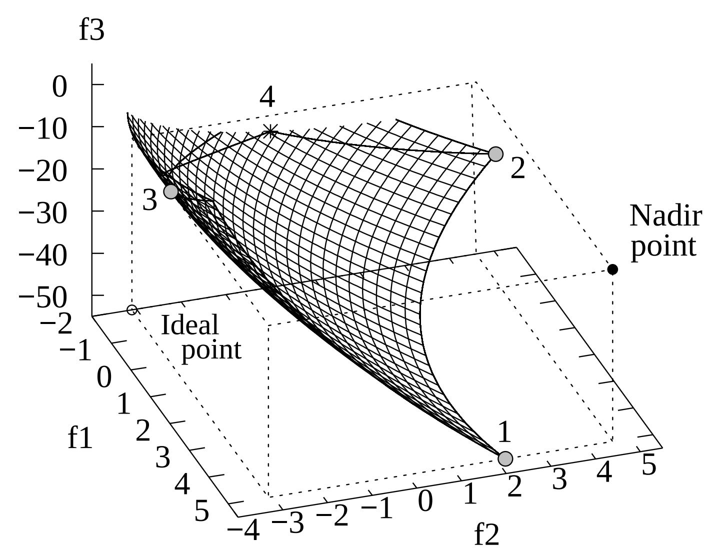

Fig. 18. Pareto-optimal front with extreme points for problem KM.

The extremized crowded NSGA-II approach is terminated when the normalized distance metric does not change by an amount $\Delta = 0.001$ in a consecutive $\tau = 500$ generations. Figure 17 shows the variation of the normalized distance metric value computed using the above-mentioned ideal and worst objective vectors. At the end of Step 2, the estimated nadir point is $\mathbf{z}^{\rm nad} = (5, 4.6, -14.212)^T$, which seems to disagree on the third objective value with that found by the grid-search strategy.

To investigate if any further improvement is possible, we proceed to Step 3 and apply three local searches, each started with one of the first three solutions presented in the table, as these three solutions constitute the nadir point. The weight vector ($\mathbf{w}$, constructed with $\epsilon = 0.001$), the corresponding reference point ($\mathbf{z}$), and the solution of the local search for each of the three extreme points are tabulated in Table V. The extremized crowded NSGA-II solution (column 3 in the table) is used as the starting solution and `fmincon` routine (an SQP method in which every approximated quadratic programming problem is solved using the BFGS quasi-Newton method [25]) of MATLAB is used with $\rho = 10^{-7}$. The table clearly shows that solution 2 (the objective vector $(0.015, -3.1, -14.212)^T$, obtained by the extremized crowded NSGA-II), was not a Pareto-optimal solution. The local search approach starting from this solution is able to find a better solution $(0, -3.1, -14.25)^T$. This shows the importance of employing the local search approach. However, the other two extreme solutions obtained by the extremized crowded NSGA-II could not be improved further. Figure 18 shows the Pareto-optimal front for this problem. These three extreme Pareto-optimal points are marked on the front with a shaded circle. The fourth point is also shown with a star. The nadir point estimated by the combination of extremized crowded NSGA-II and the local searches is $(5, 4.6, -14.25)^T$, which is identical to that obtained by the grid search strategy [24].

B. Welded Beam Design Optimization

So far, we have applied the nadir point estimation procedure to academic test problems. They have given us confidence in our suggested procedure. Next, we consider an engineering design problem having three objectives.

This problem is a well-studied one [15], [26] having four design variables, $\mathbf{x} = (h, \ell, t, b)^T$ (dimensions specifying the welded beam). Minimizations of cost of fabrication, end deflection and normal stress are of importance in this problem. There are five non-linear constraints involving shear stress, normal stress, a physical property, buckling limitation, and end deflection. The mathematical description of the problem

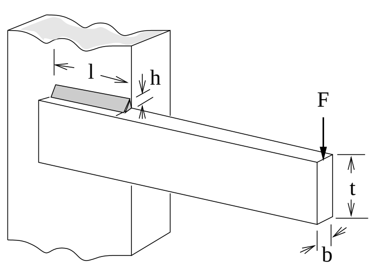

Fig. 19. The welded beam design problem.

is given below:

$$\begin{array}{ll} \text{Minimize} & \left\{ \begin{array}{l} f_1(\mathbf{x}) = 1.10471h^2\ell + 0.04811tb(14.0+\ell) \\ f_2(\mathbf{x}) = \delta(\mathbf{x}) = \frac{2.1952}{t^3 b} \\ f_3(\mathbf{x}) = \sigma(\mathbf{x}) = \frac{504{,}000}{t^2 b} \end{array} \right\}, \\ \text{Subject to} & g_1(\mathbf{x}) \equiv 13{,}600 - \tau(\mathbf{x}) \geq 0, \\ & g_2(\mathbf{x}) \equiv 30{,}000 - \sigma(\mathbf{x}) \geq 0, \\ & g_3(\mathbf{x}) \equiv b - h \geq 0, \\ & g_4(\mathbf{x}) \equiv P_c(\mathbf{x}) - 6{,}000 \geq 0, \\ & g_5(\mathbf{x}) \equiv 0.25 - \delta(\mathbf{x}) \geq 0, \\ & 0.125 \leq \ell, t \leq 10, \\ & 0.125 \leq h, b \leq 5, \end{array} \tag{8}$$

where the terms $\tau(\mathbf{x})$ and $P_c(\mathbf{x})$ are given as

$$\begin{aligned} \tau(\mathbf{x}) &= \left[(\tau'(\mathbf{x}))^2 + (\tau''(\mathbf{x}))^2 + \ell\tau'(\mathbf{x})\tau''(\mathbf{x})/ \right. \\ &\qquad \left. \sqrt{0.25(\ell^2+(h+t)^2)}\right]^{1/2}, \\ P_c(\mathbf{x}) &= 64{,}746.022(1-0.0282346t)tb^3. \end{aligned}$$

where

$$\begin{aligned} \tau'(\mathbf{x}) &= \frac{6{,}000}{\sqrt{2}h\ell}, \\ \tau''(\mathbf{x}) &= \frac{6{,}000(14+0.5\ell)\sqrt{0.25(\ell^2+(h+t)^2)}}{2\left[0.707h\ell(\ell^2/12+0.25(h+t)^2)\right]}. \end{aligned}$$

In this problem, we have no knowledge on the ideal and worst objective values. Since these values will be required in computing the normalized distance metric value for termination of the extremized crowded NSGA-II, we first compute them here.

1) Step 1: Computing Ideal and Worst Objective Vectors: We minimize and maximize each of three objectives to find the individual extreme points of the feasible objective space. For this purpose, we have used a single-objective real-parameter genetic algorithm with the SBX recombination and the polynomial mutation operators [20], [15]. We use the following parameter values: population size = 100, maximum generations = 500, recombination probability = 0.9, mutation probability = 0.1, distribution index for recombination = 2, and distribution index for mutation = 20. After a solution is obtained by a GA simulation, it is attempted to improve by a local search (LS) approach. Table VI shows the corresponding extreme objective values before and after the local search approaches. Interestingly, the use of the local search improves the cost objective from 2.3848 to 2.3810. As a outcome of the above single-objective optimization tasks, we obtain the ideal and worst objective values, as shown below:

	Cost	Deflection	Stress
Ideal	2.3810	0.000439	1008
Worst	333.9095	0.0713	30000

2) Step 2: Applying Extremized Crowded NSGA-II: First, we apply the extremized crowded NSGA-II approach with an identical parameter settings as used above, except that for SBX recombination $\eta_c = 10$ is used, according to the recommendation in [15] for multi-objective optimization. A termination criterion on normalized distance (ND) metric computed with ideal and worst objective vectors found above and with $\tau = 500$ and $\Delta = 0.001$ is employed. Figure 20 shows the variation of the ND metric with generation. It is interesting to note how the normalized distance metric, starting from a small value (meaning that the estimated nadir point is closer to the worst objective vector), reaches a stabilized quantity of 0.5394. Since the extremized crowded NSGA-II approach does not change the above normalized distance value for a consecutive $\tau = 500$ generations from 844 generations within a margin of $\Delta = 0.001$, the algorithm is terminated at generation number 1344.

Interestingly, only two non-dominated extreme points are found by the extremized crowded NSGA-II. They are shown in Table VII. From these two solutions, the estimated nadir point is $(36.4277, 0.0193, 26800)^T$.

TABLE VI

MINIMUM AND MAXIMUM OBJECTIVE VALUES OF THREE OBJECTIVES. THE VALUES MARKED WITH A (*) FOR VARIABLES x_1 AND x_2 CAN TAKE OTHER VALUES WITHOUT ANY CHANGE IN THE OPTIMAL OBJECTIVE VALUE AND WITHOUT MAKING THE OVERALL SOLUTION INFEASIBLE.

	Cost	Deflection	Stress	x_1	x_2	x_3	x_4
Minimum	2.3848			0.2428	6.2664	8.2972	0.2443
Min. after LS	2.3810			0.2444	6.2175	8.2915	0.2444
Maximum	333.9095			5	10	10	5
Max. after LS	333.9095			5	10	10	5
Minimum		0.000439		(*)4.4855	(*)9.5683	10	5
Min. after LS		0.000439		(*)4.4855	(*)9.5683	10	5
Maximum		0.0713		0.8071	5.0508	1.8330	5
Max. after LS		0.0713		0.8071	5.0508	1.8330	5
Minimum			1008	(*)4.5959	(*)9.9493	10	5
Min. after LS			1008	(*)4.5959	(*)9.9493	10	5
Maximum			30000	2.7294	5.7934	2.3255	3.1066
Max. after LS			30000	0.7301	5.0376	2.3308	3.0925

TABLE VII

TWO POPULATION MEMBERS OBTAINED USING THE EXTREMIZED CROWDED NSGA-II APPROACH.

Sol. No.	Cost	Deflection	Stress	x_1	x_2	x_3	x_4
Extremized crowded NSGA-II							
1.	36.4277	0.000439	1008	1.8679	0.4394	10	5
2.	3.5638	0.0193	26800	0.5161	3.2978	6.0345	0.5164
After local search							
1.	36.4209	0.000439	1008	1.7345	0.4789	10	5
2.	2.3810	0.0158	30000	0.2444	6.2175	8.2915	0.2444

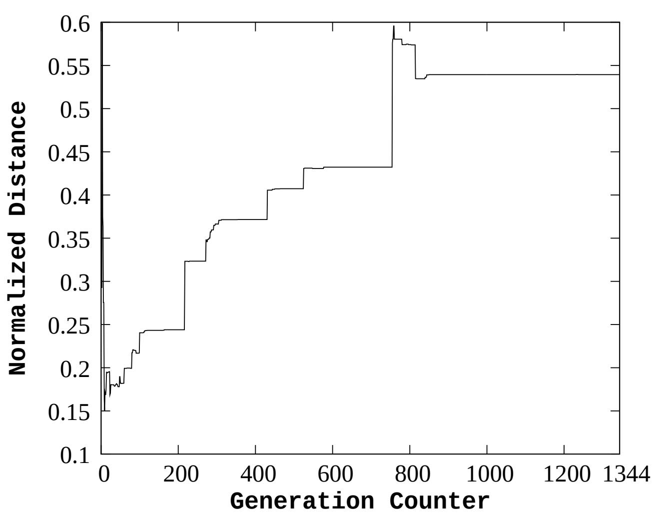

Fig. 20. Normalized distance metric till termination.

3) Step 3: Applying Local Searches: The two solutions obtained are now attempted to be improved by the local search approach, one at a time. We compute the pseudo-weight vector for each of the two solutions using equation (4). We observe that for the two obtained solutions in Table VII, the following extreme objective values result:

$$f_1^{\min} = 3.5638, \quad f_1^{\max} = 36.4277,$$
$$f_2^{\min} = 0.000439, \quad f_2^{\max} = 0.0193,$$
$$f_3^{\min} = 1008, \quad f_3^{\max} = 26800.$$

When these values are substituted in equation (4) for solution 1 and using $\epsilon = 0.001$, we obtain $\bar{\mathbf{w}}^{\mathbf{x}^{(1)}} = (1, 0.001, 0.001)^T$. For solution 2, we obtain $\bar{\mathbf{w}}^{\mathbf{x}^{(2)}} = (0.001, 0.5, 0.5)^T$. The corresponding reference points are $(52.860, 0.00439, 1008)^T$ and $(3.5638, 0.02873, 39696)^T$, respectively. We have used $\rho = 10^{-7}$ for the augmented achievement scalarizing function. We now apply `fmincon` routine of MATLAB to find the optimal solutions corresponding to the augmented achievement scalarizing functions. The solutions obtained are shown in Table VII with a heading 'After local search'.

Interestingly, the local search improves the first solution to a slightly better one. However, for the second solution, the local search finds a non-dominated solution which is better in terms of the first two objectives but worse in the third objective. Since this solution corresponds to the smallest cost solution of the two extremized crowded NSGA-II solutions, the weight vector and reference point are selected in a manner so as to target finding a better cost solution. The proposed local search is able to achieve this task, but at the expense of the third objective. Interestingly, this cost objective value is exactly the same as that obtained by the minimization of cost objective alone in Table VI. It is clear that the extremized NSGA-II approach in Step 2 found a solution close to an extreme Pareto-optimal solution and the application of Step 3 helps to move this solution to the extreme Pareto-optimal solution. This study clearly shows the efficacy of our suggestion of an appropriate weight vector and reference point for the local search approach.

Observing these two final solutions, we can now estimate the nadir point (cost, deflection, stress) for the welded beam design problem:

Nadir point: $(36.4209, 0.0158, 30000)^T$.

4) Verification Through the Naive NSGA-II Approach: In order to verify the estimated nadir point obtained by the proposed procedure, finally we apply the naive NSGA-II approach in which the naive NSGA-II is applied to the three-objective optimization problem to find the entire Pareto-optimal front. Thereafter, the range of the Pareto-optimal front will then provide us information about the nadir point. We use an identical parameter setting as used in the extremized crowded NSGA-II simulation. Again, we use a hybrid NSGA-II and local search procedure here. The local search approach used here is applied to a few NSGA-II solutions one at a time and is described in [15]. We employ (`fconmin`) of MATLAB for this purpose. In Figure 21, we show the NSGA-

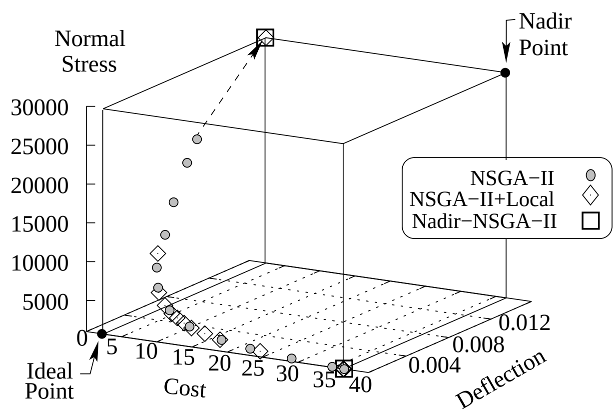

Fig. 21. Pareto-optimal front and estimation of nadir point.

II solutions with circles and their improvements by the local search method with diamonds. Two non-dominated extreme solutions obtained using our nadir point estimation procedure are marked using squares. Both approaches find an identical nadir point, thereby providing confidence to our approach proposed.

VII. Conclusions

We have proposed a hybrid methodology involving evolutionary and local search approaches for estimating the nadir point in a multi-objective optimization problem. By definition, a nadir point is constructed from the worst objective values corresponding to the solutions of the Pareto-optimal front. It has been argued that the estimation of the nadir point is an important matter in multi-objective optimization. Since the nadir point relates to the extreme Pareto-optimal solutions, the estimation of nadir point is a difficult task. Since intermediate Pareto-optimal solutions are not important in this task, the suggested NSGA-II approaches have emphasized the worst or extreme solutions corresponding to each objective. To enhance the convergence properties and make the approaches reliable, modified NSGA-II approaches are combined with a reference point based local search procedure. The extremized crowded approach has been found to be capable of making a quicker estimate of the nadir point than a naive approach (of employing the naive NSGA-II approach to first find a set of Pareto-optimal solutions and then construct the nadir point) on a number of test problems having two to 20 objectives and on a difficult engineering design problem involving non-linear objectives and constraints. In addition, we have tried the procedure to solve other numerical test problems as well. In this paper, constraints have been handled using the constraint domination principle implemented in NSGA-II. Based on the study, we can conclude the following:

1) Emphasizing both best and worst objective values in a non-dominated front has been found to be a better approach than emphasizing only the worst objective values. Since the former approach maintains a diverse set of solutions near the worst objective values and can in principle dominate spurious solutions to remain in the population, the result of the search is better and more reliable than the worst crowded approach.
2) The computational effort to estimate the nadir point has been observed to be much lower (more than an order of magnitude) for many objectives than the naive NSGA-II approach.
3) For problems with up to three objectives and for DTLZ5 problem having a low-dimensional Pareto-optimal front, both proposed approaches have been observed to perform well.

We have listed reasons for which nadir objective vectors are needed. They included normalizing objective functions, giving information about the ranges of objective functions within the Pareto-optimal front to the decision maker, visualizing Pareto-optimal solutions, and enabling the decision maker to use different interactive methods. What is common to all these is that the nadir objective vector can be computed beforehand, without involving the decision maker. Thus, it is not a problem if several hundred function evaluations are needed in the extremized crowded NSGA-II. Approximating the nadir point can be an independent task to be executed before performing any decision analysis.

One of the reasons why it may be advisable to use some interactive method for identifying the most preferred solution instead of trying to approximate the whole set of Pareto-optimal solutions is that for problems with several objectives, for example, the NSGA-II approach requires a huge number of evaluations to find a representative set. For such problems, the nadir point may be estimated quickly and reliably using the hybrid NSGA-II-cum-local-search procedure. The extremized crowded NSGA-II approach can be applied with a coarse termination requirement, so as to obtain near extreme non-dominated solutions quickly. Then, the suggested local search approach can be employed to converge to the extreme Pareto-optimal solutions reliably and accurately. Thereafter, an interactive procedure (like NIMBUS [1], for example) (using both ideal and nadir points obtained) can be applied interactively with a decision-maker to find a desired Pareto-optimal solution as the most preferred solution.

This study is important in another aspect. The proposed nadir point estimation procedure uses a hybridization of EMO and a local search based multiple criteria decision making

(MCDM) approaches. The population aspect of EMO has been used to find near extreme Pareto-optimal solutions simultaneously and the reference point based local search methodology helped converge to true extreme Pareto-optimal solutions so that the nadir point can be estimated reliably. Such collaborative EMO-MCDM studies help develop hybrid and efficient procedures which use best aspects of both contemporary fields of multi-objective optimization. Hopefully, this study will motivate researchers to engage in more such collaborative studies for the benefit of either field and, above all, to the triumph of the field of multi-objective optimization.

ACKNOWLEDGEMENTS

Authors acknowledge the FiDiPro support of the first author by the Academy of Finland.

REFERENCES

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

[2] K. Miettinen, M. M. Mäkelä, and K. Kaario, "Experiments with classification-based scalarizing functions in interactive multiobjective optimization," *European Journal of Operational Research*, vol. 175, no. 2, pp. 931–947, 2006.

[3] J. T. Buchanan, "A naive approach for solving MCDM problems: The GUESS method," *Journal of the Operational Research Society*, vol. 48, no. 2, pp. 202–206, 1997.

[4] K. Miettinen and M. M. Mäkelä, "Synchronous approach in interactive multiobjective optimization," *European Journal of Operational Research*, vol. 170, no. 3, pp. 909–922, 2006.

[5] K. Miettinen, "Graphical illustration of Pareto optimal solutions," in *Multi-Objective Programming and Goal Programming: Theory and Applications*, T. Tanino, T. Tanaka, and M. Inuiguchi, Eds. Berlin, Heidelberg: Springer-Verlag, 2003, pp. 197–202.

[6] R. Benayoun, J. de Montgolfier, J. Tergny, and P. Laritchev, "Linear programming with multiple objective functions: Step method (STEM)," *Mathematical Programming*, vol. 1, no. 3, pp. 366–375, 1971.

[7] H. Iserman and R. E. Steuer, "Computational experience concerning payoff tables and minimum criterion values over the efficient set," *European Journal of Operational Research*, vol. 33, no. 1, pp. 91–97, 1988.

[8] M. I. Dessouky, M. Ghiassi, and W. J. Davis, "Estimates of the minimum nondominated criterion values in multiple-criteria decision-making," *Engineering Costs and Production Economics*, vol. 10, pp. 95–104, 1986.

[9] P. Korhonen, S. Salo, and R. Steuer, "A heuristic for estimating nadir criterion values in multiple objective linear programming," *Operations Research*, vol. 45, no. 5, pp. 751–757, 1997.

[10] M. Ehrgott and D. Tenfelde-Podehl, "Computation of ideal and nadir values and implications for their use in MCDM methods," *European Journal of Operational Research*, vol. 151, pp. 119–139, 2003.

[11] M. Szczepanski and A. P. Wierzbicki, "Application of multiple crieterion evolutionary algorithm to vector optimization, decision support and reference point approaches," *Journal of Telecommunications and Information Technology*, vol. 3, pp. 16–33, 2003.

[12] H. P. Benson, "Existence of efficient solutions for vector maximization problems," *Journal of Optimization Theory and Applications*, vol. 26, no. 4, pp. 569–580, 1978.

[13] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

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

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

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

[17] K. Deb and D. Saxena, "Searching for pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems," in *Proceedings of the World Congress on Computational Intelligence (WCCI-2006)*, 2006, pp. 3352–3360.

[18] D. W. Corne and J. D. Knowles, "Techniques for highly multiobjective optimisation: some nondominated points are better than others," in *GECCO'07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*. New York, NY, USA: ACM Press, 2007, pp. 773–780.

[19] K. Deb and A. Srinivasan, "Innovization: Innovating design principles through optimization." in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2006)*, New York: The Association of Computing Machinery (ACM), 2006, pp. 1629–1636.

[20] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.

[21] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation Journal*, vol. 8, no. 2, pp. 125–148, 2000.

[22] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multi-objective optimization," in *Evolutionary Multi-objective Optimization*, A. Abraham, L. Jain, and R. Goldberg, Eds. London: Springer-Verlag, 2005, pp. 105–145.

[23] K. Klamroth and K. Miettinen, "Integrating approximation and interactibe decision making in multicrietria optimization," *Operations Research*, in press, dOI 10.1287/opre.1070.0425.

[24] P. Eskelinen, 2007, personal Communication.

[25] S. S. Rao, "Genetic algorithmic approach for multiobjective optimization of structures," in *Proceedings of the ASME Annual Winter Meeting on Structures and Controls Optimization*, vol. 38, 1993, pp. 29–38.

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