Nadir Point Estimation Using Evolutionary Approaches: Better Accuracy and Computational Speed Through Focused Search

Kalyanmoy Deb and Kaisa Miettinen

Abstract Estimation of the nadir objective vector representing worst objective function values in the set of Pareto-optimal solutions is an important task, particularly for multi-objective optimization problems having more than two conflicting objectives. Along with the ideal point, nadir point can be used to normalize the objectives so that multi-objective optimization algorithms can be used more reliably. The knowledge of the nadir point is also a pre-requisite to many multiple criteria decision making methodologies. Moreover, nadir point is useful for an aid in interactive methodologies and visualization softwares catered for multi-objective optimization. However, the computation of an exact nadir point for more than two objectives is not an easy matter, simply because the nadir point demands the knowledge of extreme Pareto-optimal solutions. In the past few years, researchers have proposed several nadir point estimation procedures using evolutionary optimization methodologies. In this paper, we review the past studies and reveal an interesting chronicle of events in this direction. To make the estimation procedure computationally faster and more accurate, the methodologies were refined one after the other by mainly focusing on finding smaller and still sufficient subset of Pareto-optimal solutions to facilitate estimating the nadir point. Simulation results on a number of numerical test problems demonstrate better efficacy of the approach which aims to find only the extreme Pareto-optimal points compared to other two approaches.

Keywords Nadir point · Multiobjective optimization · Evolutionary multiobjective optimization · EMO · Bilevel optimization · Payoff table · Hybrid search

K. Deb (✉)

Department of Mechanical Engineering, Indian Institute of Technology Kanpur

PIN 208016, India,

e-mail: deb@iitk.ac.in.

Also Finland Distinguished Professor,

Helsinki School of Economics, Finland

1 Introduction

A reliable and accurate estimation of the nadir point in multi-objective optimization representing the worst objective function values in the set of Pareto optimal solutions is an important task for a number of reasons. First, along with the ideal objective vector, the nadir objective vector (i.e., the nadir point) can be used to *normalize* objective functions (Miettinen 1999), a matter often desired for an adequate functioning of multi-objective optimization algorithms in the presence of objective functions with different magnitudes. Second, the nadir objective vector is a pre-requisite for finding preferred Pareto-optimal solutions in different interactive algorithms, such as the *guess* method (Buchanan 1997) (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 (Miettinen 1999; Miettinen and Mäkelä 2006). Third, 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.

Despite the long-term efforts by researchers, estimation of a nadir point is still an open matter for research. Recently, researchers have suggested different ways to employ an evolutionary multi-objective optimization (EMO) procedure for this purpose. Since an EMO methodology works with a number of points in each iteration, its operators can be designed to focus its search towards a number of Pareto-optimal solutions simultaneously in a single simulation. This flexibility makes an EMO procedure a potential tool for arriving at important Pareto-optimal points for estimating the nadir point. In this paper, we review the existing EMO methodologies from the point of view of amount of information to be calculated and discuss advantages and disadvantages of these methodologies. We argue that an EMO method of finding the extreme Pareto-optimal points (instead of all Pareto-optimal points) is a computationally faster approach and the modification of obtained extreme points by a local search may provide accuracy in the estimation of the nadir point.

Motivations for estimating the nadir point led the MCDM researchers to suggest procedures for approximating the nadir point using a so-called *payoff table* (Benayoun et al. 1971) in 1971. 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, Isermann and Steuer (1988) 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. (1986) suggested three heuristic methods and Korhonen et al. (1997) 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 are linear functions of the variables.

In Ehrgott and Tenfelde-Podehl (2003), an algorithm for deriving the nadir point was proposed based on subproblems. In other words, in order to find the nadir point for an M -objective problem, Pareto-optimal solutions of all $\binom{M}{2}$ bi-objective optimization problems must first be found. Such a requirement may make the algorithm computationally impractical beyond three objectives, although Szczepanski and Wierzbicki (2003) implemented the above idea using evolutionary algorithms (EAs) and showed applications up to three and four objective linear optimization 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 (Benson 1978).

Since an estimation of the nadir objective vector necessitates information about the whole Pareto-optimal surface, any procedure of estimating this point should involve finding Pareto-optimal solutions. This makes the task more difficult compared to finding other important points, such as the ideal point (Korhonen et al. 1997). Since EMO algorithms are potential for finding an approximate set of the entire or a part of the Pareto-optimal surface (Deb 2001), they stand as viable candidates for this task. A couple of recent studies (Deb et al. 2006, 2009; Szczepanski and Wierzbicki 2003) have demonstrated a promise in this direction. Another motivation for using an EMO procedure is that nadir point estimation is to be made only once in a problem before beginning the actual decision making process. So, even if the proposed procedure uses somewhat large computational effort (one of the criticisms made often against evolutionary optimization methods), a reliable and accurate methodology for estimating the nadir point is desired.

2 Nadir Point Estimation Procedures with Accuracy and Computational Time

In this section, we review the existing evolutionary optimization based nadir point estimation procedures from a point of view of achieving greater accuracy and a faster computational approach in estimating the nadir point. The nadir point can be estimated from any of the following scenarios: (a) the entire Pareto-optimal surface is known, (b) the *critical edges* of the Pareto-optimal surface (boundaries of Pareto-optimal surface responsible for locating the nadir point) are known, or (c) only the *critical extreme* Pareto-optimal points (extreme points of the Pareto-optimal surface responsible for locating the nadir point) are known. Interestingly, at least one procedure for each of the above tasks has already been suggested and we describe them here. However, as the amount of information to be generated is reduced from approaches (a) to (c), the computational time to estimate the nadir point are expected to be better.

2.1 Surface-to-Nadir: Computing Solutions from Entire Pareto-optimal Surface

A naive and simple-minded idea comes from finding a representative set of the entire Pareto-optimal surface with an EMO approach. Although the idea is intuitive, the difficulties of this method are many: (a) it needs an exponentially higher number of points to cover the entire Pareto-optimal surface as the number of objectives increase, (b) to estimate the nadir point accurately, it must find extreme Pareto-optimal points accurately, thereby deserving special attention for the search of the extreme points, (c) it often requires a diversity parameter specifying the minimum desirable distance between any two obtained points, hence making the procedure sensitive to a parameter. In addition, Pareto-optimality cannot be guaranteed by an EMO, but only solutions that are not dominated by other solutions in the current population are obtained. They are referred to as non-dominated points. An earlier study (Szczepanski and Wierzbicki 2003) has shown the effect of the diversity parameter on the obtained accuracy of the estimated nadir point. Further, EMO methodologies have shown to not work well in finding a well-distributed set of solutions on the entire Pareto-optimal surface for more than four objectives, thereby making EMO methodologies difficult to apply in practical scenarios.

2.2 Edge-to-Nadir: Computing Edge Solutions of Pareto-optimal Surface

Since intermediate (non-extreme) Pareto-optimal solutions do not usually contribute in determining the location of the nadir point, one may try to find only critical edges (boundaries responsible for a true estimate of the nadir point) of the Pareto-optimal surface. One way to do this would be to solve $\binom{M}{2}$ pair-wise objective combinations and collect the corresponding solutions together by computing the missing objectives (Ehrgott and Tenfelde-Podehl 2003; Szczepanski and Wierzbicki 2003). The dominated points can then be deleted and the nadir point can be estimated. However, although this procedure requires a relatively smaller computational effort than that in the surface-to-nadir approach, there are still some difficulties: (a) the accuracy of the procedure depends on the diversity parameter used to find a distributed set of solutions on the critical edges of the Pareto-optimal surface, (b) many pair-wise optimizations may find the same boundary (or a part of it) repeatedly, thereby wasting computational efforts, (c) such a technique may require to find Pareto-optimal solutions (having Pareto optimal solutions with identical values for $f_i - f_j$ but differing in at least one other objective, for example), and may need to employ a lexicographic procedure to find the true extreme Pareto-optimal points.

2.3 *Extreme-point-to-Nadir: Computing Objective-wise Worst Pareto-optimal Points*

It is intuitive to realize that even most of the intermediate edge points are not required to estimate the nadir point. It is then quite tempting to develop an optimization procedure which will target on finding only the extreme Pareto-optimal points needed to locate the nadir point accurately. A couple of recent studies (Deb et al. 2006, 2009; Szczepanski and Wierzbicki 2003) suggested such procedures using an EMO approach and showed their potential for the purpose. Since only a few points are the target in this approach, the procedure is also expected to be computationally efficient. Due to its potential to be faster and more efficient, we discuss this procedure in somewhat more detail.

2.3.1 **Extremized Crowded NSGA-II**

In the original elitist non-dominated sorted GA (NSGA-II) procedure (Deb et al. 2002), elite, non-dominated, less-crowded population members are emphasized in a systematic manner so that at the end of a simulation a well-converged and well-diversified set of non-dominated solutions estimating Pareto-optimal solutions is obtained. In the extremized crowded NSGA-II, the emphasis is placed in locating objective-wise extreme Pareto-optimal solutions so that the final population contains near objective-wise best and worst solutions. For this purpose, although non-dominated and elite solutions are still emphasized, the original crowding mechanism is modified (Deb et al. 2006, 2009).

However, evolutionary optimization algorithms are approximate search procedures and are not guaranteed to find the exact extreme solutions in every simulation. Consider Fig. 1, in which we demonstrate this scenario by showing an extreme NSGA-II solution as $\mathbf{z}_{\text{EMO}}$. Say, this solution, among all other obtained extreme NSGA-II solutions, is worst for objective f_j . Notice that it is not the exact extreme solution $\mathbf{z}_E$. Thus, to locate the nadir point accurately, a hybrid approach is desired, in which extreme NSGA-II solutions can be modified using a local search procedure, so that solutions like $\mathbf{z}_{\text{EMO}}$ can be modified to $\mathbf{z}_E$. A previous study (Deb et al. 2009) suggested a bilevel optimization procedure for this task. In this study, we make that procedure more efficient by reducing the size of the variable set used in the local search procedure.

2.3.2 **Proposed Local Search Approach**

As mentioned earlier, the extreme NSGA-II solution (such as $\mathbf{z}_{\text{EMO}}$) is often a near-extreme solution, but need not even be Pareto-optimal. To modify such solutions to reach the true extreme solution (such as $\mathbf{z}_E$), there are two tasks that must be accomplished simultaneously. First, the resulting solution must be a Pareto-optimal

Fig. 1 The local search procedure is illustrated

solution and second, the solution must have the worst value corresponding to the underlying objective. Each of these tasks is an optimization problem itself and is inter-connected to each other. To achieve the first task, we would need an optimization procedure which guarantees finding a Pareto-optimal solution. For this purpose, in our earlier study (Deb et al. 2009), we employed the achievement (scalarizing) function (ASF) (Wierzbicki 1980), which requires a reference point ($\mathbf{z}$) and a weight vector ($\mathbf{w}$) to be specified (see Fig. 1). When the achievement function is optimized, $\mathbf{z}$ is projected to become Pareto-optimal ($\mathbf{z}_P$). Since this solution is not necessarily the extreme Pareto-optimal solution $\mathbf{z}_E$ and it depends on the chosen $\mathbf{z}$ and $\mathbf{w}$ vectors, we needed an upper-level optimization task in which these two vectors ($\mathbf{z}$ and $\mathbf{w}$) were used as decision variables (Deb et al. 2009). The j -th objective value of $\mathbf{z}_P$ was considered as the objective of the upper-level optimization problem (for each $j = 1, \dots, M$ in turn).

Thus, the working of the bilevel local search procedure was as follows. The upper-level optimization task performed a search in the space of reference point and weight vectors. A specific solution ($\mathbf{z}, \mathbf{w}$) was evaluated by solving a lower-level optimization problem. The lower-level optimization performed a search in the space of original decision variables ($\mathbf{x}$) and solved the ASF function with the specific ($\mathbf{z}, \mathbf{w}$) vectors to find the corresponding Pareto-optimal solution $\mathbf{z}_P$. The j -th objective function value of this solution was then assigned as the objective value of the upper-level solution ($\mathbf{z}, \mathbf{w}$). Since the lower-level task guarantees a Pareto-optimal solution and the upper-level task works for maximizing corresponding f_j value, the end result is expected to be the extreme Pareto-optimal solution $\mathbf{z}_E$ for the j -th objective function.

Here, we make the above bilevel local search procedure more efficient by considering only the reference point ($\mathbf{z}$) as decision variable vector in the upper level optimization task and use a weight vector determined by the worst and best currently

known objective values for every objective f_i . This is actually enough to guarantee finding any Pareto-optimal solution (Miettinen 1999).

2.3.3 Hybrid Extreme-point-to-nadir Algorithm

Now we are ready to describe the overall hybrid extreme-point-to-nadir algorithm.

- Step 1: Compute ideal ($\mathbf{z}^*$) and worst ($\mathbf{z}^w$) objective vectors of the entire search space by minimizing and maximizing each function individually. They are needed in computing the termination criterion for the extremized crowded NSGA-II procedure.
- Step 2: Apply extremized-crowded NSGA-II approach (Deb et al. 2006, 2009) to find a set of non-dominated extreme points. The estimated nadir point ($\mathbf{z}^{\text{est}}$) is then computed for checking the termination of the procedure. Iterations are continued till a condition on the normalized distance metric (e.g., $\Delta ND \leq 0.0001$) is met, where

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

and the normalized change in the distance metric ($\Delta ND = (ND_{\max} - ND_{\min}) / ND_{\text{avg}}$) is noted for the past 50 generations for checking the termination condition. At the end of this simulation, P non-dominated near-extreme points ($\mathbf{x}_{\text{EMO}}^{(j)}$ for $j = 1, 2, \dots, P$) are found. Thereafter, we identify the best and worst objective vectors $\mathbf{f}^{\min}$ and $\mathbf{f}^{\max}$ from these P solutions.

- Step 3: Apply the bilevel local search procedure from each near-extreme solution $\mathbf{x}_{\text{EMO}}^{(j)}$ (using the corresponding objective vector $\mathbf{z}_{\text{EMO}} = \mathbf{f}_{\text{EMO}}^{(j)}$) to locate the true extreme solution $\mathbf{y}^{*(j)}$. The upper-level optimization uses the reference point ($\mathbf{z}$) as the decision variable vector and maximizes an objective function which is determined by solving a lower-level optimization problem (given in 3):

$$\begin{aligned} \text{(Upper-level problem)} \quad & \begin{cases} \text{Maximize}_{(\mathbf{z})} & f_k^{*(j)}(\mathbf{z}), \\ \text{subject to} & \mathbf{z} \in \mathbf{f}_{\text{EMO}}^{(j)} + [-0.5, 1.5](\mathbf{f}^{\max} - \mathbf{f}^{\min}). \end{cases} \end{aligned} \quad (2)$$

The objective function $f_k^{*(j)}(\mathbf{z})$ is the value of k -th objective function computed at the optimal solution $\mathbf{y}^{*(j)}$ of the following lower-level optimization problem, in which the vector $\mathbf{z}$ is used as the reference point for the augmented achievement scalarizing function:

$$\begin{aligned} \text{(Lower-level problem)} \quad & \begin{cases} \text{Minimize}_{(\mathbf{y}^{(j)})} & \max_{i=1}^M \left(\frac{f_i(\mathbf{y}^{(j)}) - z_i}{f_i^{\max} - f_i^{\min}} \right) + \rho \sum_{m=1}^M \left(\frac{f_m(\mathbf{y}^{(j)}) - z_m}{f_m^{\max} - f_m^{\min}} \right), \\ \text{subject to} & \mathbf{y}^{(j)} \in \mathcal{S}, \end{cases} \end{aligned} \quad (3)$$

where $\mathcal{S}$ is the feasible search space restricted by the original constraints. The upper-level optimization is initialized with the NSGA-II solution $\mathbf{z}_{(0)} = \mathbf{f}_{\text{EMO}}^{(j)}$. The lower-level optimization is initialized with the NSGA-II solution $\mathbf{y}_{(0)}^{(j)} = \mathbf{x}_{\text{EMO}}^{(j)}$. The resulting optimal solution for the bilevel local search is $\mathbf{y}_{\text{LS}}^{(j)}$ with an objective vector $\mathbf{f}_{\text{LS}}^{(j)}$. Step 3 is repeated for all those of the P NSGA-II solutions that contain worst values for some objectives (denoted by j in Step 3).

Step 4: Finally, construct the nadir point from the worst objective values of extreme Pareto-optimal points ($\mathbf{f}_{\text{LS}}^{(j)}$), $j = 1, 2, \dots, P$) obtained by the local search procedure.

The use of augmented achievement scalarizing function does not allow the lower-level optimization to converge to a weak Pareto-optimal solution. But, in certain problems, the approach may only allow to find an extreme *proper* Pareto-optimal solution (Miettinen 1999) depending on the value of the parameter ρ . Alternatively, it is possible to use a lexicographic formulation of the achievement scalarizing function to guarantee Pareto optimality (Miettinen 1999).

3 Results on Numerical Test Problems

In this section, we present results of test runs with the hybrid extreme-point-to-nadir algorithm and compare its performance to the other two procedures for which results are borrowed from the original study (Szczepanski and Wierzbicki 2003). For all runs involving the NSGA-II procedure, we have used the following parameter values. Population size (N) is proportional to number of variables (n), as $N = 20n$. Crossover and mutation probabilities are 0.9 and $1/n$, respectively. The distribution index for the simulated binary crossover operator (SBX) (Deb 2001) is 10 and the same for polynomial mutation operator (Deb 2001) is 50. The parameter ρ for the augmented scalarizing function is set to 10^{-5} . Multiple runs were made to test the sensitivity of the algorithms on different initial populations, but for space restrictions, we present the results of a typical run.

3.1 Problem SZ1

We borrow the first two problems from Szczepanski and Wierzbicki (2003) which applied the first two nadir point estimation methodologies (surface-to-nadir and edge-to-nadir approaches). The first problem is as follows:

$$\text{Min } \left\{ \begin{array}{l} f_1(\mathbf{x}) = -(100 - 7x_1 - 20x_2 - 9x_3), \\ f_2(\mathbf{x}) = -(4x_1 + 5x_2 + 3x_3), \\ f_3(\mathbf{x}) = -x_3, \end{array} \right\}, \quad \text{s.t. } \begin{array}{l} 1\frac{1}{2}x_1 + x_2 + 1\frac{3}{5}x_3 \leq 9, \\ x_1 + 2x_2 + x_3 \leq 10, \\ x_i \geq 0, \quad i = 1, 2, 3. \end{array} \quad (4)$$

Fig. 2 Feasible objective space and Pareto-optimal surface for problem SZ1

The study Szczepanski and Wierzbicki (2003) reported the true nadir point to be $\mathbf{z}^{\text{nad}} = (-3.6364, 0, 0)^T$. Figure 2 shows a sketch of the feasible objective space and the corresponding Pareto-optimal surface (shaded region).

The surface-to-nadir approach first finds a set of well-distributed points on the entire Pareto-optimal surface and then constructs the nadir point from the obtained points. After 120,000 *solution evaluations* (total number of solutions evaluated during the entire optimization process), the point $(-5.06, 0, 0)^T$ was declared as an estimate of the nadir point in Szczepanski and Wierzbicki (2003). The edge-to-nadir approach finds the Pareto-optimal edges corresponding to pair-wise minimizations of objectives. In this problem, all three pairs of objectives will find representative points on the edges shown with a thick line in the figure. The nadir point is then estimated to be $(-4.78, 0, 0)^T$ (Szczepanski and Wierzbicki 2003). Even after 360,000 solution evaluations, the resulting estimate using the edge-to-nadir approach was $(-4.36, 0, 0)^T$, having a 20% higher value in the first objective from the true value. Although the problem is linear with only three variables, the above two evolutionary methodologies seem to have faced difficulties in finding the true nadir point in this problem. We now employ the extreme-point-to-nadir approach with the proposed local search procedure.

In Step 1, we found the ideal point by minimizing each objective using MATLAB's *linprog* code, which employs the simplex method of linear programming. Three minimizations provided $\mathbf{z}^* = (-100, -31, -5.625)^T$ as the ideal point, requiring 1, 3, and 2 solution evaluations, respectively. We also needed the worst point for terminating Step 2. The point $\mathbf{z}^w = (0, 0, 0)^T$ was found with 4, 1, and 1 solution evaluations, respectively. A total of 12 solutions evaluations were needed to compute $\mathbf{z}^*$ and $\mathbf{z}^w$.

In Step 2, we applied the extremized crowded NSGA-II with a parameter setting as described above. Figure 3 shows the variation of the normalized distance metric with generation. The algorithm was terminated after generation 108 and the total amount of solution evaluations needed in Steps 1 and 2 was $60 \times 109 + 12 = 6,552$. Four solutions were found in Step 2 and they are presented in Table 1 and represented in Fig. 2 with diamonds. Notice, how the modified NSGA-II finds non-dominated near-extreme points (A, B, C, and D) for the entire Pareto-optimal surface, without finding any intermediate points.

Fig. 3 Normalized distance metric for problem SZ1

Table 1 Four solutions found by the extremized crowded NSGA-II for Problem SZ1. ‘LS’ stands for results after local search performed on solutions found worst in terms of the k -th objective function

j	$\mathbf{f}_{\text{EMO}}^{(j)}$	$\mathbf{x}_{\text{EMO}}^{(j)}$	k	$\mathbf{f}_{\text{LS}}^{(j)}$
1	$(-12.4541, -30.9050, -0.0052)^T$	$(4.006, 2.973, 0.0053)^T$	–	Not a worst extreme point
2	$(-49.3921, -16.8739, -5.6207)^T$	$(0.003, 0.000, 5.621)^T$	–	Not a worst extreme point
3	$(-3.7878, -26.8347, -3.5788)^T$	$(0.043, 3.185, 3.579)^T$	1	$(-3.6364, -26.8182, -3.6364)^T$
4	$(-100.0000, 0.0000, 0.0000)^T$	$(0.000, 0.000, 0.000)^T$	2,3	$(-100, 0, 0)^T$

From the table with four obtained solutions, we observe the worst and best objective vectors as $\mathbf{f}^{\max} = (-3.7878, 0.0000, 0.0000)^T$ and $\mathbf{f}^{\min} = (-100.0000, -30.9050, -5.6207)^T$, respectively. Interesting to note that $\mathbf{f}^{\max}$ is already close to the true nadir point $\mathbf{z}^{\text{nad}} = (-3.6364, 0, 0)^T$. Thereafter, we employed the local search procedure of Step 3 from the two solutions corresponding to the worst objective vector $\mathbf{f}^{\max}$ to investigate if we are able to find the nadir point more accurately. Since the problem is linear, we employed the linear programming code linprog of MATLAB software for the lower-level optimization task.

In Step 3, we observe that solution 1 (point D in Fig. 2) and solution 2 (point B) do not contribute to worst objective values, so we ignore these points from further consideration. In fact, these two solutions exist in the NSGA-II final population because they correspond to the minimum value of objectives f_2 and f_3 , respectively. Solution 3 (point C) corresponds to the worst of objective f_1 (with $k = 1$) and hence will be subjected to a local search in the hope of improving it to reach the true extreme (worst) Pareto-optimal point corresponding to objective f_1 . The resulting solution (point O) is shown in the table. The bilevel optimization requires 712 solution evaluations.

Next, we consider solution 4 (point A), which corresponds to the worst of both objectives f_2 and f_3 (with $k = 2$ and 3). Thus, we maximize a normalized sum of both these objectives $\left(\sum_{j=2}^3 \frac{(f_j(\mathbf{x}) - f_j^{\min})}{f_j^{\max} - f_j^{\min}} \right)$ in the inner-loop of local search method. The same point $\mathbf{f}_{\text{LS}}^{(4)} = (-100, 0, 0)^T$ is found in only 83 solution evaluations.

In Step 4, we collate these points $(\mathbf{f}_{\text{LS}}^{(3)})$ and $(\mathbf{f}_{\text{LS}}^{(4)})$ and declare the estimated nadir point as $(-3.6364, 0, 0)^T$, which is identical to the exact nadir point. The total

number of solution evaluations needed by all steps of the procedure was 7,347, of which about 10.8% was needed by the local search procedure. The computation needed by this hybrid extreme-point-to-nadir approach is only about 6.1% of that needed by the quicker of the other two approaches and importantly the extreme-point-to-nadir approach also finds a more accurate result. This study demonstrates how the task of finding the nadir point can become computationally faster and accurate if the focus is made in finding extreme points, rather than on the entire Pareto-optimal surface or on the edges of the Pareto-optimal surface.

3.2 Problem SZ2

Next, we consider another problem from Szczepanski and Wierzbicki (2003):

$$\text{Min} \left\{ \begin{array}{l} 9x_1 + 19.5x_2 + 7.5x_3 \\ 7x_1 + 20x_2 + 9x_3 \\ -4x_1 - 5x_2 - 3x_3 \\ -x_3 \end{array} \right\}, \quad \text{s.t.} \quad \begin{array}{l} 1.5x_1 + x_2 + 1.6x_3 \leq 9, \\ x_1 + 2x_2 + x_3 \leq 10, \\ x_i \geq 0, \quad i = 1, 2, 3. \end{array} \quad (5)$$

The true nadir point for this problem is $\mathbf{z}^{\text{nad}} = (94.5, 96.3636, 0, 0)^T$. The study Szczepanski and Wierzbicki (2003) obtained a close point $(94.4998, 95.8747, 0, 0)^T$ using the edge-to-nadir approach. This study required a total of 120,000 solution evaluations. To compare the surface-to-nadir approach, we applied the original NSGA-II procedure to obtain a set of 200 points (requiring a total of 200,000 solution evaluations). The nadir point estimated from the obtained solution set was $(94.4031, 96.2597, 0, 0)^T$, which is not quite the true nadir point for this problem.

Now, we discuss the results of extreme-point-to-nadir approach on this problem. In Step 1 of the procedure, we found the ideal and worst objectives values: $\mathbf{z}^* = (0, 0, -31, -5.625)^T$ and $\mathbf{z}^w = (97.5, 100, 0, 0)^T$, respectively. This required $(1 + 1 + 3 + 2) = 7$ and $(4 + 4 + 1 + 1) = 10$ solution evaluations, respectively.

Thereafter, in Step 2, we applied the extremized crowded NSGA-II procedure using a population size of 60 and initializing the population around $x_i \in [0, 10]$ for all three variables. The NSGA-II run was terminated at generation 315 with the prescribed termination criterion, thereby requiring a total of $50 \times 316 + 17$ or 12,657 solution evaluations in Steps 1 and 2. Solutions obtained are tabulated in Table 2.

The minimum and maximum objective vectors (found by linprog routine of MATLAB software) are: $\mathbf{f}^{\text{min}} = (0.0000, 0.0000, -30.9920, -5.6249)^T$ and $\mathbf{f}^{\text{max}} = (94.4810, 96.3635, 0.0000, 0.0000)^T$, respectively. Notice that the maximum vector is close to the true nadir point mentioned above. We shall now investigate whether the proposed local search is able to improve this point to find the exact nadir point.

We observe that the objective values of the first solution does not correspond to any element of $\mathbf{f}^{\text{max}}$. Thus, in Step 3, we employed the bilevel local search procedure only for the other three solutions. The resulting solutions and corresponding $\mathbf{z}_{\text{LS}}^{(j)}$

Table 2 Extremized crowded NSGA-II and local search method on problem SZ2

j	$\mathbf{x}_{\text{EMO}}^{(j)}$	Objective vector, $\mathbf{f}_{\text{EMO}}^{(j)}$	
1	$(0.0001, 0, 5.6249)^T$	$(42.1879, 50.6249, -16.8752, -5.6249)^T$	
2	$(0.0001, 3.1830, 3.6336)^T$	$(89.3219, 96.3635, -26.8164, -3.6336)^T$	
3	$(3.9980, 2.9998, 0.0003)^T$	$(94.4810, 87.9854, -30.9920, -0.0003)^T$	
4	$(0, 0, 0)^T$	$(0, 0, 0, 0)^T$	
j	k	$\mathbf{z}_{\text{LS}}^{(j)}$	Extreme point, $\mathbf{f}_{\text{LS}}^{(j)}$
1	–	No worst objective value	
2	2	$(231.0, 96.4, -42.3, -6.4)^T$	$(89.3182, 96.3636, -26.8182, -3.6364)^T$
3	1	$(236.2, 88.0, -46.5, 8.4)^T$	$(94.5000, 88.0000, -31.0000, 0.0000)^T$
4	3,4	$(0, 0, 0, 0)^T$	$(0, 0, 0, 0)^T$

vectors are shown in the table. For solutions 2 and 3, we maximize objectives f_2 and f_1 , respectively. Since solution 4 is worst with respect to both objectives f_3 and f_4 , we maximize the sum of normalized objectives, as described for the previous problem. The solution evaluations required till convergence for each of the three optimizations are 688, 125, and 44, respectively.

From the obtained local search solutions (last column in the table), we estimate the nadir point as $(94.5, 96.3636, 0, 0)^T$, which is identical to the true nadir point for this problem. The total number of solution evaluations was 13,514. This is only about 11.3% of the total solution evaluations needed in Szczepanski and Wierzbicki (2003). Moreover, our approach finds the exact nadir point, whereas Szczepanski and Wierzbicki (2003) could not find the exact nadir point even with about seven times more solution evaluations.

3.3 Problem IS

Next, we consider a problem presented in Isermann and Steuer (1988):

$$\begin{aligned}
 & \text{s.t. } x_1 + x_2 + 3x_3 + 3x_5 + 2x_6 \leq 61, \\
 & \quad 3x_2 + 2x_3 + 4x_4 \leq 72, \\
 & \quad 5x_1 + 3x_2 + 5x_5 + 4x_6 + 4x_7 \leq 76, \\
 & \quad 4x_1 + 2x_2 + 4x_4 + 4x_6 \leq 51, \\
 & \quad 5x_1 + 2x_2 + 3x_4 + x_5 + 4x_6 \leq 66, \\
 & \quad 2x_1 + 2x_2 + 4x_4 + 4x_5 + 4x_6 \\
 & \quad \quad + 5x_7 \leq 59, \\
 & \quad 3c_1 + 2x_3 + 5x_5 + x_6 + 2x_7 \leq 77, \\
 & \quad 0 \leq x_i \leq 20.
 \end{aligned}
 \tag{6}$$

The original study reported that the nadir point for this problem is $\mathbf{z}^{\text{nad}} = (3.50, -35.156, -28.7, -4.278)^T$. This point was difficult to find using the payoff table (Isermann and Steuer 1988). Because of four objectives, the original NSGA-II

Table 3 Extremized crowded NSGA-II and local search solutions for problem IS. Note that problem IS is a maximization problem

j	$\mathbf{f}_{\text{EMO}}^{(j)}$	k	$\mathbf{f}_{\text{LS}}^{(j)}$
1	$(-22.830, -5.886, 25.315, 25.603)^T$	1	$(3.500, -5.200, 25.400, 38.600)^T$
2	$(29.217, -34.034, 5.924, 15.766)^T$	2	$(31.563, -35.156, 10.563, 9.219)^T$
3	$(40.024, 5.651, -27.052, 29.768)^T$	3	$(47.567, 12.750, -28.700, 16.633)^T$
4	$(43.483, -14.710, -8.610, -4.954)^T$	4	$(49.167, -27.528, -9.389, -4.278)^T$
5	$(48.178, -22.520, -9.045, -0.717)^T$	–	Not a worst extreme point
6	$(16.955, 44.198, -0.032, 1.101)^T$	–	Not a worst extreme point
7	$(4.145, -5.945, 25.490, 39.174)^T$	–	Not a worst extreme point
8	$(5.281, -14.513, 4.162, 57.196)^T$	–	Not a worst extreme point

approach may not work well in this problem in finding the extreme points precisely. When testing the surface-to-nadir approach, with 200 population members in each iteration, the NSGA-II after 200,000 function evaluations produced the following nadir point: $\mathbf{z}^{\text{est}} = (-25.067, -35.032, -28.623, -6.877)^T$, which is not close to the true nadir point, especially on the first objective.

With four objectives, the edge-to-nadir approach needs $\binom{4}{2}$ or 6 bi-objective optimization runs. Employing NSGA-II for this purpose with 140 population members running for 35,000 solution evaluations each required a total of 210,000 solution evaluations. The resulting estimated nadir point was $\mathbf{z}^{\text{est}} = (-9.339, -23.600, -25.497, 0.069)^T$. Due to the inherent difficulties of not finding the appropriate solutions in the bi-objective optimizations, discussed elsewhere (Deb et al. 2009), the procedure cannot estimate the nadir point properly.

Now, we apply the hybrid extreme-point-to-nadir approach. The minimum and maximum objective values of the entire search space are found to be $\mathbf{f}^{\min} = (-49.167, -44.250, -25.500, -59.609)^T$ and $\mathbf{f}^{\max} = (29.114, 45.500, 28.700, 20.000)^T$. These tasks require a total of $17 + 13$ or 30 solution evaluations by the linear programming routine (linprog) of the MATLAB software. It is clear that $\mathbf{f}^{\max}$ is not close to the nadir point of this problem. Next, we apply the extremized crowded NSGA-II with a population size of 140. Eight extreme solutions were obtained and are presented in Table 3. The optimization required a total of 39,900 solution evaluations. Thereafter, we employed the bilevel local search from each of the four worst extreme solutions. The resulting solutions are also presented in the table. The solution evaluations needed were 4,887, 2,424, 6,052, and 250, respectively. The estimated nadir point from the final column of the table is $\mathbf{z}^{\text{est}} = (3.500, -35.156, -28.700, -4.278)^T$, which is identical to the true nadir point. The overall amount of solution evaluations needed was 53,543, of which about 25% was needed by the local search procedure. Even in this four-objective problem, the hybrid extreme-point-to-nadir approach was more accurate and significantly less computationally demanding in estimating the nadir point than the other two competing approaches.

3.4 Problem KM

Finally, we consider a non-linear problem, which provides difficulty for the payoff table method. This problem was used in Klamroth and Miettinen (2008):

$$\text{Min } \left\{ \begin{array}{l} -x_1 - x_2 + 5 \\ \frac{1}{5}(x_1^2 - 10x_1 + x_2^2 - 4x_2 + 11) \\ (5 - x_1)(x_2 - 11) \end{array} \right\}, \quad \text{s.t. } \begin{array}{l} 3x_1 + x_2 - 12 \leq 0, \\ 2x_1 + x_2 - 9 \leq 0, \\ x_1 + 2x_2 - 12 \leq 0, \\ 0 \leq x_1 \leq 4, \quad 0 \leq x_2 \leq 6. \end{array} \quad (7)$$

The payoff table method finds $(5, 2.2, -14.25)^T$ as the wrongly estimated nadir point. Another study (Eskelinen et al. 2008) as well as a grid-search strategy (computationally possible due to the presence of only three objectives) of creating a number of feasible solutions systematically and constructing the nadir point from the solutions obtained estimated the nadir point to be $(5, 4.6, -14.25)^T$.

For this problem, previous studies using surface-to-nadir or edge-to-nadir approaches are not available. Thus, we simulate the two procedures here. The surface-to-nadir approach using the original NSGA-II procedure finds $(5.0, 4.60, -14.225)^T$ after 100,000 solution evaluations. The edge-to-nadir approach using bi-objective NSGA-II simulations estimate the nadir point as $(5.0, 4.46, -14.2088)^T$ after a total of 180,000 solution evaluations. Both these estimated points are different from that obtained by the grid search strategy.

We now employ the extreme-point-to-nadir approach. In Step 1, individual minimizations of objectives (using nonlinear optimization solver `fmincon` of MATLAB) identify the vector $\mathbf{z}^* = (-2, -3.1, -55)^T$ as the ideal objective vector. This required a total of $(18 + 37 + 9) = 64$ solution evaluations. The maximization of the objectives led to the worst objective vector $\mathbf{z}^w = (5, 4.6, -14.25)^T$ with $(12 + 18 + 18) = 48$ solution evaluations.

In Step 2 of the approach, we employ the extremized crowded NSGA-II and found four non-dominated extreme solutions, as shown in the second column of Table 4. The simulation was terminated after 135 generations, thereby requiring $40 \times 136 + 112 = 5,552$ solution evaluations in Steps 1 and 2. At the end of Step 2, the estimated nadir point is $(5, 4.6, -14.212)^T$, which seems to disagree somewhat 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

Table 4 Extremized crowded NSGA-II and local search method on Problem KM

j	$\mathbf{x}_{\text{EMO}}^{(j)}$	Objective vector, $\mathbf{f}_{\text{EMO}}^{(j)}$	k	Extreme point, $\mathbf{f}_{\text{LS}}^{(j)}$
1	$(0, 0)^T$	$(5, 2.2, -55)^T$	1	$(5, 2.2, -55)^T$
2	$(3.511, 1.466)^T$	$(0.023, -3.100, -14.194)^T$	3	$(0, -3.1, -14.25)^T$
3	$(0, 6)^T$	$(-1, 4.6, -25)^T$	2	$(-1, 4.6, -25)^T$
4	$(2.007, 4.965)^T$	$(-1.973, -0.050, -18.060)^T$	–	Not a worst extreme point

in Table 4, as these three solutions correspond to the worst value of one of the objectives. The minimum and maximum objective vectors from these solutions are: $\mathbf{f}^{\min} = (-1, -3.1, -55)^T$ and $\mathbf{f}^{\max} = (5, 4.6, -14.194)^T$, respectively. Solution 1 from the table corresponds to the worst value of the first objective ($k = 1$). Thus, the upper-level optimization run maximizes objective f_1 . This optimization took 122 solution evaluations. The table clearly shows that solution 2 obtained by NSGA-II was not a Pareto-optimal point. The local search approach starting from this solution is able to find a better solution $(0, -3.1, -14.25)^T$, requiring a total of 408 solution evaluations. This shows the importance of employing the local search approach in obtaining exact extreme points. The third solution could not be improved any further, since it is already the desired extreme point with respect to f_2 with $k = 2$, but the optimization requires 112 solution evaluations to terminate with the prescribed conditions.

The nadir point estimated from last column of Table 4 is $(5, 4.6, -14.25)^T$, which is identical to that obtained by the grid search strategy (Eskelinen et al. 2008). Overall, the hybrid extreme-point-to-nadir approach required 6,194 solution evaluations to estimate the nadir point exactly to this non-linear problem, for which the local search procedure required about 10.4% of the total computations. The exact nadir point was obtained by this approach with only about 6% solution evaluations compared to the quicker of the other two approaches.

4 Conclusions

Recent studies have shown that EMO procedures are potential for the estimation of a nadir point. In this paper, we have reviewed three such implementations which seemed to vary according to the amount of computation needed to estimate the nadir point. By comparing the number of solution evaluations of these procedures, we have concluded that the extreme-point-to-nadir approach which directly focuses to find extreme Pareto-optimal points not only estimates the nadir point more accurately, but also requires an order of magnitude less solution evaluations compared to other two approaches. Similar results have been observed on other problems (which we could not provide here due to space restrictions). However, the bilevel approach used here is a nested approach which was found to be computationally somewhat expensive in some cases. Recent efficient coevolutionary approaches for bilevel optimization (Deb and Sinha 2009) can be applied for a faster computation of the extreme points. The hybrid extreme-point-to-nadir approach seems to be a promising procedure for making a reliable and accurate estimate of the nadir point in linear and non-linear multi-objective optimization problems.

Acknowledgements Authors acknowledge the FiDiPro support from the Academy of Finland (grant 118319). The research was also partially funded by the Jenny and Antti Wihuri Foundation.

References

- Benayoun, R., de Montgolfier, J., Tergny, J., & Laritchev, O. (1971). Linear programming with multiple objective functions: Step method (STEM). *Mathematical Programming*, 1(3), 366–375.
- Benson, H. P. (1978). Existence of efficient solutions for vector maximization problems. *Journal of Optimization Theory and Applications*, 26(4), 569–580.
- Buchanan, J. T. (1997). A naive approach for solving MCDM problems: The GUESS method. *Journal of the Operational Research Society*, 48(2), 202–206.
- Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley.
- Deb, K. & Sinha, A. (2009). Solving bilevel multi-objective optimization problems using evolutionary algorithms. In *Proceedings of Evolutionary Multi-Criterion Optimization (EMO-2009)* (pp. 110–124). Heidelberg: Springer.
- Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197.
- Deb, K., Chaudhuri, S., & Miettinen, K. (2006). Towards estimating nadir objective vector using evolutionary approaches. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2006)* (pp. 643–650). New York: The Association of Computing Machinery (ACM).
- Deb, K., Miettinen, K., & Chaudhuri, S. (2009). An estimation of nadir objective vector using a hybrid evolutionary-cum-local-search procedure. Technical report, Helsinki School of Economics, Finland. Working Paper W-470.
- Dessouky, M. I., Ghiassi, M., & Davis, W. J. (1986). Estimates of the minimum nondominated criterion values in multiple-criteria decision-making. *Engineering Costs and Production Economics*, 10, 95–104.
- Ehrgott, M. & Tenfelde-Podehl, D. (2003). Computation of ideal and nadir values and implications for their use in MCDM methods. *European Journal of Operational Research*, 151, 119–139.
- Eskelinen, P., Miettinen, K., Klamroth, K., & Hakanen, J. (2008). Pareto Navigator for interactive nonlinear multiobjective optimization. *OR Spectrum*. DOI 10.1007/s00291-008-0151-6
- Isermann, H. & Steuer, R. E. (1988). Computational experience concerning payoff tables and minimum criterion values over the efficient set. *European Journal of Operational Research*, 33(1), 91–97.
- Klamroth, K. & Miettinen, K. (2008). Integrating approximation and interactive decision making in multicriteria optimization. *Operations Research*, 56, 222–234.
- Korhonen, P., Salo, S., & Steuer, R. (1997). A heuristic for estimating nadir criterion values in multiple objective linear programming. *Operations Research*, 45(5), 751–757.
- Miettinen, K. (1999). *Nonlinear Multiobjective Optimization*. Boston: Kluwer.
- Miettinen, K. & Mäkelä, M. M. (2006). Synchronous approach in interactive multiobjective optimization. *European Journal of Operational Research*, 170(3), 909–922.
- Szczepanski, M. & Wierzbicki, A. P. (2003). Application of multiple criterion evolutionary algorithm to vector optimization, decision support and reference point approaches. *Journal of Telecommunications and Information Technology*, 3, 16–33.
- Wierzbicki, A. P. (1980). The use of reference objectives in multiobjective optimization. In G. Fandel & T. Gal (Eds.), *Multiple Criteria Decision Making Theory and Applications* (pp. 468–486). Berlin: Springer.