# This is an electronic reprint of the original article.
# This reprint *may differ* from the original in pagination and typographic detail.

**Author(s):** Ruiz, Ana B.; Sindhya, Karthik; Miettinen, Kaisa; Ruiz, Francisco; Luque, Mariano

**Title:** E-NAUTILUS: A decision support system for complex multiobjective optimization problems based on the NAUTILUS method

**Year:** 2015

**Version:**

# E-NAUTILUS: A Decision Support System for Complex Multiobjective Optimization Problems based on the NAUTILUS Method

**Ana B. Ruiz** (abruiz@uma.es)
Department of Applied Economics (Mathematics),
Universidad de Málaga, Calle Ejido 6, 29071 Málaga, Spain
**Karthik Sindhya** (karthik.sindhya@jyu.fi) and
**Kaisa Miettinen** (kaisa.miettinen@jyu.fi)
University of Jyvaskyla, Department of Mathematical Information Technology,
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland
**Francisco Ruiz** (rua@uma.es) and
**Mariano Luque** (mluque@uma.es)
Department of Applied Economics (Mathematics),
Universidad de Málaga, Calle Ejido 6, 29071 Málaga, Spain

**Abstract**

Interactive multiobjective optimization methods cannot necessarily be easily used when (industrial) multiobjective optimization problems are involved. There are at least two important factors to be considered with any interactive method: computationally expensive functions and aspects of human behavior. In this paper, we propose a method based on the existing NAUTILUS method and call it the *Enhanced NAUTILUS (E-NAUTILUS)* method. This method borrows the motivation of NAUTILUS along with the human aspects related to avoiding trading-off and anchoring bias and extends its applicability for computationally expensive multiobjective optimization problems. In the E-NAUTILUS method, a set of Pareto optimal solutions is calculated in a pre-processing stage before the decision maker is involved. When the decision maker interacts with the solution process in the interactive decision making stage, no new optimization problem is solved, thus, avoiding the waiting time for the decision maker to obtain new solutions according to her/his preferences. In this stage, starting from the worst possible objective function values, the decision maker is shown a set of points in the objective space, from which (s)he chooses one as the preferable point. At successive iterations, (s)he always sees points which improve all the objective values achieved by the previously chosen point. In this way, the decision maker remains focused on the solution process, as there is no loss in any objective function value between successive iterations. The last post-processing stage ensures the Pareto optimality of the final solution. A real-life engineering problem is used to demonstrate how E-NAUTILUS works in practice.

**Keywords:** Multiple objective programming, Interactive methods, Multiple criteria optimization, Computational cost, Trading-off, Pareto optimality.

# 1  Introduction

Optimization problems, e.g., in industry, often involve multiple conflicting objectives and such problems are typically referred to as multiobjective optimization problems. Such problems are usually complex as they deal with nonlinear objective and constraint functions, different types of variables and may involve the use of computationally expensive black-boxes simulators which imitate some physical processes. Moreover, multiobjective optimization problems do not have a single optimal solution but, instead, several (usually infinitely many) optimal solutions, called Pareto optimal solutions, with different trade-offs among the objectives. Because of this, there is a need for an expert, referred to as the decision maker (DM), who can provide her/his preference information when needed and who ultimately chooses one solution among several others based on her/his preferences. In the literature, there exists a plethora of methods to handle multiobjective optimization problems and among them interactive methods are commonly used (1; 2). In interactive methods, the DM is actively involved during the iterative solution process. Here, the preference information of the DM is accounted to generate a set of solutions which are subsequently presented to her/him. Next (s)he investigates and chooses one or more solutions as the preferred solution(s). This process continues for several iterations until the DM is satisfied with the solution found and regards it as the most preferred Pareto optimal solution.

Interactive methods are bound by aspects of human behavior, such as the fact that losses loom larger than gains and the susceptibility of a DM to an anchoring bias, as mentioned by Kahneman and Tversky in (3). According to them, in an anchoring bias, the DM defines an adaptation level based on her/his past and present context of experience and subsequently her/his response is in relation to her/his adaptation level. On the other hand, moving from one Pareto optimal solution to another requires sacrifice(s) in some objective function(s), which may hinder the DM from finding desirable solutions. Both these aspects would mean that the starting solution(s) will have a profound impact on the final solution chosen by the DM. Though these aspects of human behavior are critical in understanding the behavior of a DM, very few interactive methods consider them. In a recently proposed interactive method suggested in (4), the NAUTILUS method, special emphasis is devoted to the different aspects of human behavior laid down by Kahneman and Tversky. In NAUTILUS, the DM starts from the worst possible objective function values derived from the set of Pareto optimal solutions and progressively moves towards her/his preferred Pareto optimal solution, by improving every objective function at the current iteration as compared to the previous one. In this way, one can make sure that the DM does not have to trade off at any iteration and, in fact, (s)he gains on each of the objectives.

However, the NAUTILUS method involves a large computational cost. At every iteration of NAUTILUS, a new single objective optimization problem has to be solved unless the DM does not provide any new preference information. In addition, a number of single objective optimization problems equal to the number of objectives has to be solved in order to find the bounds of the reachable objective function values at any iteration, which are provided to the DM. For complex multiobjective optimization problems involving computationally expensive objective and constraint functions, solving multiple single objective optimization problems at every iteration is not practical as the DM typically has to wait for a considerable time period dictated by the time taken to evaluate the objective and constraint functions between iterations. In this paper, we address this discrepancy in the NAUTILUS method with several enhancements and we call the result as the *Enhanced-NAUTILUS (E-NAUTILUS)* method.

In a nutshell, the E-NAUTILUS method applies the three-stage solution process ideology

proposed in (5) and consists of the following three stages: pre-processing, interactive decision making and post-processing stages. In order to reduce the computing time during the interactive decision making stage, a well-spread set of Pareto optimal solutions is pre-computed using e.g. any a posteriori type method (1) or evolutionary multiobjective optimization algorithms (6; 7). Evolutionary multiobjective optimization algorithms are nature inspired population based algorithms, where a set of random solutions is considered to start with and which evolves over several generations to finally obtain a set of solutions as close as possible to the set of Pareto optimal solutions. It must be noted that, since Pareto optimal solutions cannot be guaranteed when using evolutionary multiobjective optimization optimization algorithms, additional efforts must be devoted to ensure the optimality of the solutions, see e.g. (8). If a well-spread set of solutions cannot be obtained e.g. because of a need for an exponentially large number of solutions to represent a high-dimensional objective space, approximation techniques such as the PAINT method (9), the interactive hybrid approach (10) etc. can be used to generate new solutions.

In the interactive decision making stage, the original computationally expensive problem is not solved but the calculation is based on the data generated in the pre-processing stage. The DM is iteratively shown a set of intermediate points at a certain distance from the nadir point (defined in Section 2) in the objective space, from which (s)he selects one point. This set of intermediate points represents the different ways of approaching the Pareto optimal front from the previous point selected, i.e., different ways of improving the current objective function values. At every iteration, these intermediate points are closer to the set of Pareto optimal solutions and farther from the nadir point than the ones obtained in the previous iteration. This interactive stage is terminated when the DM chooses one solution among the solutions calculated in the pre-processing stage. Initially, the DM specifies the number of intermediate points (s)he wishes to investigate at every iteration and the number of iterations (s)he would like to pursue before her/his preferred Pareto optimal solution is reached. Based on the number of iterations specified, step lengths are calculated to reach the set of Pareto optimal solutions and, at each iteration, a step is taken assuring that the solutions generated are closer to the set of Pareto optimal solutions. In addition, the DM never looses sight of the set of Pareto optimal solutions since, at every iteration, (s)he is provided information about the reachable part of the Pareto optimal set from every intermediate point. It must be noted that the intermediate points only guide the DM towards her/his preferred Pareto optimal solution, but these points are not Pareto optimal, and they may not even be feasible. Finally, the Pareto optimality of the preferred solution chosen by the DM is ensured in the post-processing stage. As mentioned earlier, the Pareto optimality of the solutions generated in the pre-processing stage cannot necessarily be guaranteed and, thus, the last stage is needed.

One can regard the E-NAUTILUS method as an interactive decision support system for the DM to find the most preferred solution among the ones generated by any a posteriori method. It supports the DM as only a small amount of solutions is considered at a time and it progressively gives her/him an idea of what can and cannot be achieved in the problem, without the necessity of evaluating the objective functions of the problem again and again. State-of-the-art of hybrid evolutionary multiobjective optimization algorithms has been surveyed in (11). As stated in (11), there are several a posteriori algorithms which incorporate preferences of the DM once an approximation of the Pareto optimal front is found. However, none of them is similar to E-NAUTILUS.

Next, in Section 2, we present the background concepts which lay a foundation to the rest of the article. In this section, we also present a brief description of the NAUTILUS method, which

is the backbone of the E-NAUTILUS method. In Section 3, we give a detailed description of the E-NAUTILUS method and, in Section 4, we demostrate the E-NAUTILUS method using a real-life three-objective optimization problem associated with the efficiency improvement of the auxiliary services of thermal power plants. Finally, we conclude and give future research directions in Section 5.

## 2  Background Concepts

We consider *multiobjective optimization problems* of the form

$$
\begin{aligned}
\text{minimize} \quad & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\} \\
\text{subject to} \quad & \mathbf{x} \in S,
\end{aligned}
\tag{1}
$$

with $k \geq 2$ conflicting objective functions $f_i : S \to \mathbb{R}$. The *decision vectors* $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ belong to the feasible set $S \subset \mathbb{R}^n$. For every $\mathbf{x} \in S$, we denote the vector of objective function values by $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^T$, which is called an *objective vector* in the *objective space* $\mathbb{R}^k$. Usually, because of the conflict degree among the objectives, it is not possible to find a solution where all the objectives can reach their individual optimum, so we are interested in so-called *Pareto optimal solutions*, at which no objective function value can be improved without impairing, at least, one of the others. Given $\mathbf{z}^1, \mathbf{z}^2 \in \mathbb{R}^k$, we say that $\mathbf{z}^1$ *dominates* $\mathbf{z}^2$ if $z_i^1 \leq z_i^2$ for all $i = 1, 2, \ldots, k$ and $z_j^1 < z_j^2$ for at least one index $j$. If $\mathbf{z}^1$ and $\mathbf{z}^2$ do not dominate each other, we say that $\mathbf{z}^1$ and $\mathbf{z}^2$ are *(mutually) nondominated*. Then, we say that a decision vector $\mathbf{x}^* \in S$ is *Pareto optimal* if there does not exist another $\mathbf{x} \in S$ such that $\mathbf{f}(\mathbf{x})$ dominates $\mathbf{f}(\mathbf{x}^*)$. The corresponding objective vector $\mathbf{f}(\mathbf{x}^*)$ is called a *Pareto optimal objective vector*. Usually, problem (1) has many Pareto optimal solutions. The set formed by all Pareto optimal solutions is called the *Pareto optimal set*, denoted by $E$, and its image in the objective space is referred to as the *Pareto optimal front*, denoted by $\mathbf{f}(E)$. In this paper, we refer to objective vectors which map with decision vectors as *solutions*, and to vectors in the objective space which do not necessarily correspond to any decision vector as *points*. Besides, in this paper, for a point $\mathbf{z} \in \mathbb{R}^k$, we say that a solution $\mathbf{x} \in S$, is *reachable from* $\mathbf{z}$ if $\mathbf{f}(\mathbf{x})$ dominates $\mathbf{z}$, that is, if $f_i(\mathbf{x}) \leq z_i$ for every $i = 1, \ldots, k$.

When solving problem(s), the DM decides which Pareto optimal solution satisfies best her/his preferences. This solution is commonly known as *the most preferred solution* and is defined as the final solution of the problem. We will denote it as $\mathbf{z}_{pref}$. There are different ways of expressing preferences (1; 12; 13), such as weighting the objective functions, expressing local trade-offs for the objectives, performing pairwise comparisons, selecting one solution among a set of solutions or giving a reference point formed by desirable objective functions values, among others. One of the ways is to provide to the DM a set of solutions and to let her/him choose the one (s)he likes most. In E-NAUTILUS, the DM expresses her/his preferences in this way.

The ranges of the objective function values in the Pareto optimal front are defined by the ideal and the nadir points (also known as the ideal and nadir objective vectors), which represent the best and the worst values that each objective function can achieve in the Pareto optimal set, respectively. The *ideal point*, $\mathbf{z}^\star = (z_1^\star, \ldots, z_k^\star)^T$, contains the lowest objective function values and is obtained by $z_i^\star = \min_{\mathbf{x} \in S} f_i(\mathbf{x}) = \min_{\mathbf{x} \in E} f_i(\mathbf{x})$ for all $i = 1, \ldots, k$. The *nadir point*, $\mathbf{z}^{nad} = (z_1^{nad}, \ldots, z_k^{nad})^T$, is formed by the highest objective function values and can be defined as $z_i^{nad} = \max_{\mathbf{x} \in E} f_i(\mathbf{x})$ for all $i = 1, \ldots, k$. In practice, the nadir point is usually difficult

to calculate as the set $E$ is unknown and we need to approximate it (see (1) and references therein). A pay-off table (14) has often been used to find an approximation of the nadir point, but lately more reliable approaches for its estimation have been suggested (see, e.g., (15; 16)). Alternatively, the DM can also be asked for the worst possible objective function values and such values can be used as the components of the nadir point.

As previously mentioned, E-NAUTILUS is based on the NAUTILUS method proposed by Miettinen et al. in (4). NAUTILUS is an interactive method and was specially designed for avoiding the undesired anchoring effects that many interactive methods suffer from. In many of them, the necessity of an impairment in at least one objective function to find a new solution may hinder the decision making task and this may imply that the DM focusses on a subset of Pareto optimal solutions too prematurely. As suggested in (3; 17), past experiences affect people's hopes and expectations, and DMs do not always react in the same way to gains and losses. These properties may make the selection of a satisfactory solution difficult or even endanger finding the most preferred solution. Because of that, NAUTILUS starts the search for the most preferred solution from the worst objective function values, that is, from the nadir point. From it, the search is oriented in such a way that the DM will progressively focus on the part of the Pareto optimal front (s)he likes most. In fact, the DM can reach any Pareto optimal solution with the NAUTILUS method without the necessity to sacrifice, i.e., trade off. Although only the last solution will be Pareto optimal, the DM is always shown a solution which dominates the solution given at the previous iteration, which means that (s)he gains an improvement in all the objectives at every iteration. This should encourage her/him to continue iterating, given that better objective function values are obtained progressively, without a need of trading off. However, as the solution process approaches the most preferred solution, the area of the Pareto optimal front that can still be reached without trading off shrinks. This implies that there will be other regions of the Pareto optimal front that will not be reachable unless a step backwards is taken.

Figure 1 gives a graphical idea of the iterative solution process in NAUTILUS for a biobjective optimization problem. At the beginning, the DM indicates how many iterations (s)he would like to take until finding her/his most preferred solution. However, the number of iterations can be changed during the solution process if (s)he desires to slow down or speed up the progress. We refer to the solution generated at each iteration $i$ as an *iteration point*, which is denoted by $\mathbf{z}^i$ in Figure 1 (for $i = 0, 1, 2$). The first iteration point is the nadir point. At each iteration, the DM expresses her/his preferences about desired local improvements of the current objective function values following the scheme proposed in (18). This information is employed to calculate a vector of preferential weights, which is subsequently used in an achievement scalarizing function (19) to find a Pareto optimal solution. From the practical point of view, these preferential weights define the search direction for the next iteration point. Actually, the next iteration point is a point lying in the segment which joins the current iteration point and the objective vector (labelled as $\mathbf{f}^i$ for $i = 1, 2$ in Figure 1) of the Pareto optimal front obtained when optimizing the achievement scalarizing function. The length of the step taken towards the Pareto optimal front is adjusted according to the number of iterations left. Additional information is also given to the DM about the ranges of the values that each objective function can reach without trading off from the next iteration point (i.e., information about the subset of Pareto optimal solutions which are reachable from the next iteration point). These ranges are given at each iteration by the current iteration point and the point $\mathbf{z}^{i+1,lo}$ in Figure 1. A measure of the proximity to the Pareto optimal front is also shown. With this information, the DM decides either to continue the search from the next iteration point in the current or in a

new direction, or to take a step backwards and redirect the search from the current iteration point if the information given is not satisfactory, by expressing new preference information. Subsequently, a new iteration is performed and this process continues until completing the number of iterations the DM wishes to take.
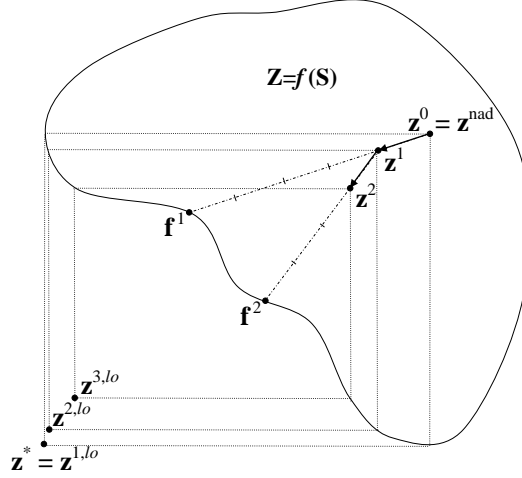


Figure 1: Graphical idea of the NAUTILUS method in a biobjective minimization problem. We can see how, for two iterations, the DM improves the objective function values, starting from the nadir point. At each iteration $i$, the direction is given by the weights obtained from the preference information provided by the DM, a corresponding Pareto optimal solution $\mathbf{f}^i$ is calculated, and a step towards $\mathbf{f}^i$ is taken (iteration point $\mathbf{z}^i$). The ranges of the reachable Pareto optimal solutions corresponding to each iteration point visualized are illustrated as rectangles.

# 3    E-NAUTILUS for Decision Support

The E-NAUTILUS method consists of three stages: pre-processing, interactive decision making and post-processing stages, as shown in the flowchart in Figure 2. In the pre-processing stage, a set of nondominated solutions $P$ approximating the whole Pareto optimal front is initially generated using e.g. an evolutionary multiobjective optimization algorithm or any other a posteriori method, and estimates of the nadir and the ideal points are calculated using $P$. Special effort should be devoted to ensuring that the solutions in $P$ are not only (mutually) nondominated but also as close as possible to the Pareto optimal front. As the name suggests, the DM is involved in the interactive decision making stage, where (s)he interactively and iteratively improves all objective function values to find the most preferred solution available. Finally, in the post-processing stage, an achievement scalarizing function is optimized over the feasible set, using the solution selected by the DM as the reference point, in order to find (if possible) a more accurate Pareto optimal solution. In practice, this means projecting the solution chosen onto the Pareto optimal front in the direction determined in the achievement scalarizing function (1; 19). The DM is not involved in the post-processing stage.

The interactive decision making stage is the main part of the E-NAUTILUS method. Here we briefly introduce this stage and a more detailed description is provided hereafter. In this stage, the DM interacts in the solution process by analysing a small set of points in the objective
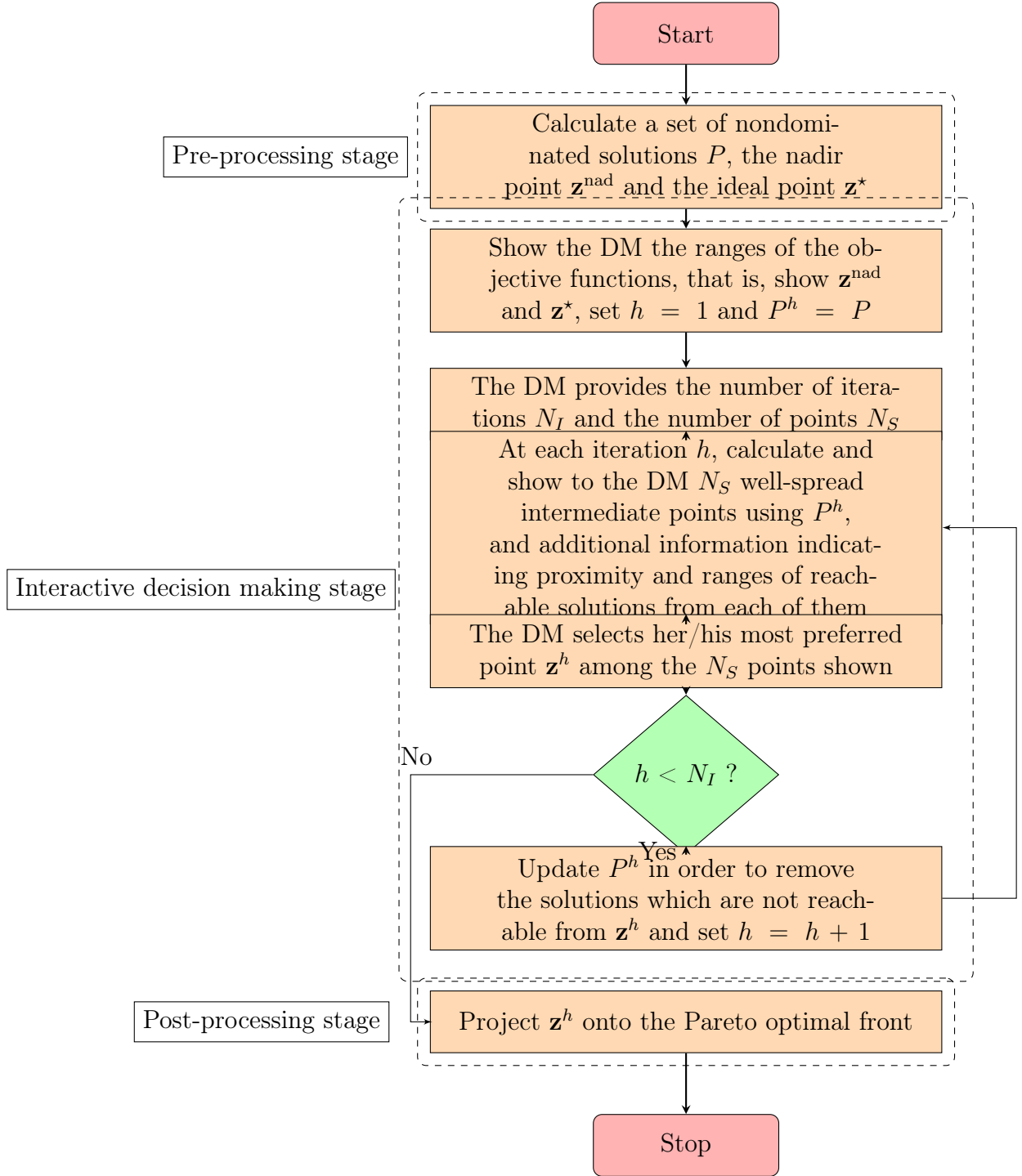
Figure 2: Three stages of the E-NAUTILUS method. The pre-processing stage is aimed at obtaining a good approximation of the whole Pareto optimal front, from which the nadir and the ideal points are estimated. In the interactive decision making stage, the DM actively participates by indicating the number of solutions to be seen and the number of iterations to be taken, and by selecting at each iteration the intermediate point (s)he likes most. Finally, once all the iterations have been carried out, the post-processing stage assures the Pareto optimality of the last point selected.

space and selecting one among them at each iteration. Before starting the interactive solution process, the ideal and the nadir points are presented to the DM in order to let her/him know the ranges of the values that each objective function can reach. Then, the DM is asked to set the number of iterations $N_I$ to be carried out and the number of points $N_S$ to investigate at each iteration. The interactive decision making stage of E-NAUTILUS starts from the nadir point, from which $N_I$ steps (or iterations) are taken towards $P$. At each iteration $h$, a set of $N_S$ mutually nondominated points, termed as *intermediate points*, are generated and shown to the DM. These $N_S$ mutually nondominated points always dominate the point chosen by the DM at the previous iteration. In order to help the DM decide which point (s)he prefers most at each iteration, information about the closeness to $P^h$ (a subset of nondominated solutions in $P$ reachable at iteration $h$) and the objective function bounds of the reachable solutions in $P^h$ (for each of the $N_S$ points) are also provided. At each iteration, the DM selects her/his most preferred point, referred to as $\mathbf{z}^h$, from the $N_S$ points shown and a new iteration is performed. This iterative procedure is repeated by providing new sets of $N_S$ well-spread mutually nondominated points to the DM until $N_I$ iterations are completed. At each iteration, except in the last one, the set $P^h$ is updated by removing those solutions which cannot be reached from $\mathbf{z}^h$. In this stage, we can assure that the point chosen by the DM at any iteration will always dominate the points (s)he has selected at the previous iterations. Thus, at each iteration, an improvement is always obtained in each objective function. This is expected to encourage the DM to continue iterating. In this way, the DM progressively moves towards a subset of the set of nondominated solutions $P$ generated in the pre-processing stage, starting from the nadir point, and only at the last iteration, (s)he considers a subset of nondominated solutions from $P$ and chooses one of them as her/his most preferred solution.

After completing the interactive decision making stage, the post-processing stage is used to ensure the Pareto optimality of the final solution. We can guarantee having a Pareto optimal solution at the end of E-NAUTILUS is by projecting the last solution selected in the interactive decision making stage, $\mathbf{z}^h$, on the Pareto optimal front. For that, we can minimize the achievement scalarizing function proposed in (19) using $\mathbf{z}^h$ as a reference point, by any suitable mathematical optimization technique. But, given that this stage means to solve a single-objective optimization problem with the original (computationally expensive) functions, it can be skipped if it is regarded to be too time-consuming. Finally, the objective function values and the decision variable values of the projected or the selected solution are provided to the DM.

Although E-NAUTILUS has the same underlying philosophy as the NAUTILUS method, there are several differences, which make it specifically suitable for computationally complex multiobjective optimization problems, such as:

1. A possibility to provide more than one point in the objective space at every iteration to the DM to consider. Although it has been stated in (20) that "trade-off conflict is a major source of decisional stress", we believe that different types of DMs exist and a method must provide the possibility for a DM to choose the number of points (s)he wishes to investigate based on her/his cognitive ability. Thus, unlike the NAUTILUS method, where the DM is shown just one point at each iteration, a set of mutually nondominated points representing different trade-offs is provided to the DM at each iteration.

2. No optimization problem is solved in the second stage, that is, when the DM is involved, using the original multiobjective optimization problem, which can be computationally expensive. Instead, a set of solutions representing the Pareto optimal front is pre-calculated

in the pre-processing stage, before the DM is involved in the solution process. This prevents the DM from waiting at each iteration when solving complex multiobjective optimization problems where the generation of Pareto optimal solutions would otherwise take time.

3. Preference information asked from the DM in E-NAUTILUS is simple, i.e., the DM is asked to choose a point among a set of points. In the NAUTILUS method, the DM has to provide information about desired local improvements in the objective function values, which may be cognitively challenging in some cases.

In what follows, we describe in detail the interactive decision making stage of E-NAUTILUS, which is the core of the method. Algorithm 1 contains the main steps of this stage. One should note that this stage of E-NAUTILUS operates in the objective space. Let us denote by $h$ the current iteration number, $it^h$ the number of iterations left at each iteration (including iteration $h$), $\mathbf{z}^h$ the selected point by the DM at the iteration $h$, referred to as the *iteration point*, and $P^h$ the subset of nondominated solutions of $P$ which can be reached at iteration $h$ from the previous iteration point $\mathbf{z}^{h-1}$ (without impairing any of the objective function values).

As said before, the ideal and the nadir points are first shown to the DM in order to let her/him know the ranges of the objective functions. As the reachable solutions always belong to $P$, the ideal and the nadir points are approximated using the best and the worst objective function values represented in $P$, respectively. Next, the DM sets values for $N_I$ (number of iterations) and $N_S$ (number of points). Initially, we set the nadir point $\mathbf{z}^{\text{nad}}$ as $\mathbf{z^0}$ and the interaction with the DM starts at $h = 1$. At this moment, the number of iterations left is $it^1 = N_I$ and $P^1 = P$, which means that from the nadir point all the nondominated solutions in $P$ can be reached.

At each iteration $h$, the set of $N_S$ intermediate points is generated. For this, first, the $N_S$ most representative solutions of $P^h$ are found and then $N_S$ points are calculated which lie on the line segments that join each representative solution and the previous iteration point $\mathbf{z}^{h-1}$. For the selection of the $N_S$ most representative solutions, we need to divide $P^h$ into $N_S$ subsets and find a representative solution of each one. This can be done by any suitable methodology, such as clustering, which can be used to reduce the number of solutions in a set, maintaining the main characteristics of the trade-offs among them. After dividing $P^h$ into $N_S$ clusters using, e.g., the average linkage method (21), we can select the centroids of the clusters as the representative solutions. For $i = 1, \ldots, N_S$, let us denote these representative solutions by $\bar{\mathbf{z}}^{h,i}$.

Subsequently, we calculate and show the DM the following intermediate points:

$$\mathbf{z}^{h,i} = \frac{it^h-1}{it^h}\mathbf{z}^{h-1} + \frac{1}{it^h}\bar{\mathbf{z}}^{h,i}, \quad \text{for every } i = 1, \ldots, N_S, \tag{2}$$

and (s)he has to select one of them as the next iteration point $\mathbf{z}^h$. As can be seen, the step-length taken towards the Pareto optimal front represented by $P$ is calculated according to $it^h$, the number of iterations left. Since $it^h$ depends on the total number of iterations to be taken, the higher the number of the iterations to carry out, the smaller the step-length is and the slower the convergence to $P$ is.

If $it^h = 1$ (i.e., we are at the last iteration), we have $\mathbf{z}^{h,i} = \bar{\mathbf{z}}^{h,i}$ from (2) and, thus, the points shown to the DM at the last iteration are solutions belonging to $P^h \subset P$. This means that the point $\mathbf{z}^{N_I}$ chosen by the DM at the last iteration is a nondominated solution in $P$. If $it^h > 1$ (i.e., we are not at the last iteration), it must be noted that the points $\mathbf{z}^{h,i}$ may not be feasible solutions of the original problem. This must not be regarded as a drawback, given that

---
**Algorithm 1** Interactive decision making stage of E-NAUTILUS
___

**Input:**   • A set of nondominated solutions, $P$, approximating the Pareto optimal front.

   • Estimates of the nadir point $\mathbf{z}^{\mathrm{nad}}$ and the ideal point $\mathbf{z}^\star$ calculated using the worst and the best objective function values in $P$.

**Output:** Most preferred nondominated solution, $\mathbf{z}_{pref}$.

**Step 1: Initial preference information**. Show $\mathbf{z}^{\mathrm{nad}}$ and $\mathbf{z}^\star$ to the DM and ask her/him to indicate the number of iterations to be carried out, $N_I$, and the number of points to investigate at each iteration, $N_S$.

**Step 2: Parameter initialization**. Set $\mathbf{z}^0 = \mathbf{z}^{\mathrm{nad}}$, $h = 1$, $it^1 = N_I$ and $P^1 = P$.

**Step 3:** Select the $N_S$ most representative points in $P^h$, denoted by $\bar{\mathbf{z}}^{h,i}$ for $i = 1, \dots, N_S$.

**Step 4:** For each $i = 1, \dots, N_S$, calculate the intermediate point $\mathbf{z}^{h,i}$ in the line segment joining $\mathbf{z}^{h-1}$ and $\bar{\mathbf{z}}^{h,i}$ as follows:

$$\mathbf{z}^{h,i} = \frac{it^h - 1}{it^h}\mathbf{z}^{h-1} + \frac{1}{it^h}\bar{\mathbf{z}}^{h,i}.$$

**Step 5:** If $it^h > 1$, for $i = 1, \dots, N_S$, calculate the lower bounds of the reachable objective function values from $\mathbf{z}^{h,i}$ and the measurement of the closeness to $P$ as follows:

   • In order to calculate the lower bound objective vector, $\mathbf{f}^{h,i,lo} = (f_1^{h,i,lo}, \dots, f_k^{h,i,lo})$, for each $r = 1, \dots, k$, solve the $\epsilon$-constraint problem $(P_r^{h,i})$ given in (3) and $f_r^{h,i,lo}$ is the optimal objective function value of problem $(P_r^{h,i})$.

   • The measurement of the closeness is computed as $d^{h,i} = \frac{\|\mathbf{z}^{h,i} - \mathbf{z}^{\mathrm{nad}}\|}{\|\bar{\mathbf{z}}^{h,i} - \mathbf{z}^{\mathrm{nad}}\|} \times 100$.

**Step 6: Preferred point**:

   • If $it^h = 1$, show $\mathbf{z}^{h,i}$ for $i = 1, \dots, N_S$ to the DM. Set the point selected by the DM as $\mathbf{z}_{pref}$ and *Stop*.

   • If $it^h > 1$, show $\mathbf{z}^{h,i}$ for $i = 1, \dots, N_S$ to the DM, together with $\mathbf{f}^{h,i,lo}$ and $d^{h,i}$. Let $\mathbf{z}^h$ be the point selected by the DM.

**Step 7:** Set $P^{h+1} = \phi$. For each $\mathbf{z} \in P^h$, if $f_j^{h,l,lo} \leq z_j \leq z_j^h$ for all $j = 1, \dots, k$, set $P^{h+1} = P^{h+1} \bigcup \{\mathbf{z}\}$. Set $it^{h+1} = it^h - 1$ and $h = h + 1$. Go to Step 3.
___

the main purpose of E-NAUTILUS is to progressively improve the objective functions at each iteration until the DM reaches a satisfactory nondominated solution in $P$, without trading off. Thus, a possibly infeasible solution is dominated by a feasible one in $P$. To be more specific, the following properties hold:

**Theorem 1.** *At any iteration $h$, for any $r_1, r_2 \in \{1 \ldots, N_S\}$, points $\mathbf{z}^{h,r_1}$ and $\mathbf{z}^{h,r_2}$ do not dominate each other.*

*Proof.* See A.

**Theorem 2.** *1. At any iteration $h$ with $it^h \neq 1$, the objective vector $\mathbf{z}^{h,i}$ is an achievable[1] point dominated by $\bar{\mathbf{z}}^{h,i}$, for any $i = 1, \ldots, N_S$.*

*2. At any iteration $h$, $\mathbf{z}^h$ dominates $\mathbf{z}^{h-1}$.*

*Proof.* See A.

Theorem 2 is an extension of Theorem 1 in (4). According to claim *1* of Theorem 2, when $it^h \neq 1$, we can assure that, although the points presented to the DM at each iteration are not Pareto optimal and may be even infeasible, there always exist feasible solutions which dominate them. Furthermore, Theorem 1 means that the $N_S$ points shown at each iteration will not dominate each other, whatever the shape of the Pareto optimal front is. Additionally, claim *2* of Theorem 2 ensures that each iteration point dominates all the previous ones, and this means that the DM will always obtain an improvement in each objective function at each iteration taken.

If $it^h > 1$, from each point $\mathbf{z}^{h,i}$, the range of reachable values that each objective function can take will shrink as new iterations are carried out. This implies that there will be a region of $P^h$ that will be discarded at the following iterations. Because of that, the DM should be aware of what kind of solutions can and cannot be reached from each $\mathbf{z}^{h,i}$ before selecting the next iteration point. Thereby, we also give the DM lower bounds of the reachable values from each $\mathbf{z}^{h,i}$, and a measurement of its closeness to $P$. Note that the upper bounds of the reachable values are given by the points $\mathbf{z}^{h,i}$ themselves. On the one hand, for every $i = 1, \ldots, N_S$, the lower bounds of the reachable values from $\mathbf{z}^{h,i}$ are calculated following the same idea as in NAUTILUS. We consider the following $\epsilon$-constraint problems for $r = 1, \ldots, k$:

$$(P_r^{h,i}) \left\{ \begin{array}{ll} \text{minimize} & f_r(\mathbf{x}) \\ \text{subject to} & f_j(\mathbf{x}) \leq z_j^{h,i}, j = 1, \ldots, k, j \neq r, \\ & \mathbf{x} \in P^h. \end{array} \right. \tag{3}$$

As can be seen, instead of solving these problems over the whole feasible set, we just minimize them over $P^h$. In this way, the computation cost of solving each problem $(P_r^{h,i})$ at each iteration $h$ (a total of $k \times N_S$ problems) is significantly small, given that we just need to find the solution in $P^h$ (a finite set) with the lowest function value for each one of them. Next, the lower bound objective vector for the point $\mathbf{z}^{h,i}$, denoted by $\mathbf{f}^{h,i,lo} = (f_1^{h,i,lo}, \ldots, f_k^{h,i,lo})$ for every $i = 1, \ldots, N_S$, is defined using the optimal values of the problems $(P_r^{h,i})$. That is, for each $r = 1, \ldots, k$, the lower bound $f_r^{h,i,lo}$ for the objective $f_r$ is the optimal objective function value of problem $(P_r^{h,i})$.

On the other hand, we measure the closeness of each $\mathbf{z}^{h,i}$ to $P^h$ as follows:

$$d^{h,i} = \frac{\|\mathbf{z}^{h,i} - \mathbf{z}^{\text{nad}}\|}{\|\bar{\mathbf{z}}^{h,i} - \mathbf{z}^{\text{nad}}\|} \times 100 \quad \text{for every } i = 1, \ldots, N_S, \tag{4}$$

---

[1]We say that a point in $\mathbb{R}^k$ is *achievable* if its components can be simultaneously achieved or improved by a feasible solution.

where $\|\cdot\|$ is the $L_2$-norm. As in the NAUTILUS method, the greater $d^{h,i}$, the closer the point $\mathbf{z}^{h,i}$ is to $P^h$. This information is useful to let the DM have an idea of how fast the approach to $P^h$ would be from each of the points shown.

With all this information, the DM analyses each $\mathbf{z}^{h,i}$, together with the lower bounds of the reachable objective function values in $\mathbf{f}^{h,i,lo}$ and the measurement of closeness $d^{h,i}$. (S)he must decide which point satisfies her/his preferences best and this point will be set as the next iteration point $\mathbf{z}^h$. The process described is repeated until $N_I$ iterations have been carried out.

As said, at each iteration, once the DM has selected the next iteration point $\mathbf{z}^h$, there is a part of $P^h$ that cannot be reached at the next iteration. Because of that, the set $P^{h+1}$ to be used at the next iteration will be formed by the solutions $\mathbf{z} \in P^h$ whose objective value $z_j$ verify that $f_j^{h,l,lo} \leq z_j \leq z_j^h$ for all $j = 1, \ldots, k$. In other words, $P^{h+1}$ will be formed by the nondominated solutions in $P$ which approximate the reachable part of $P$ from $\mathbf{z}^h$.

Figure 3 gives an idea of the first and second iterations of the interactive decision making stage of E-NAUTILUS in a biobjective optimization problem. In this example, the DM wishes to perform three iterations ($N_I = 3$) and investigate six points at each iteration ($N_S = 6$). At the first iteration, we start from $\mathbf{z}^0 = \mathbf{z}^{\mathrm{nad}}$ and $P^1 = P$. This set is divided into six clusters and one representative of each cluster is selected. The six points shown to the DM are the ones encircled in the figure on the left. They have been obtained by dividing the line segments joining the representative solutions and $\mathbf{z}^0$ into three parts. Next, assuming that the DM has selected the point labelled $\mathbf{z}^1$, the process described is repeated and the second iteration is performed, as shown in the figure on the right. It can be observed that from $\mathbf{z}^1$, the subset of $P^h$ that can be reached has shrunk (shown by a dashed box) and there are some solutions in $P^h$ that have been discarded. Since one iteration has already been completed, at the second iteration, the segments joining the representative solutions and $\mathbf{z}^1$ are divided into two parts, so that the step-length is adjusted correspondingly.

The E-NAUTILUS method allows several flexibilities in the solution process, which have not been indicated in Algorithm 1, for simplicity. These flexibilities are the following:

1. In the interactive decision making stage of E-NAUTILUS, the number of points to be seen can be altered at any moment if the DM desires so. On the other hand, if the DM is not satisfied with any of the points shown at an iteration, (s)he can take a step backwards in order to re-select the iteration point. In that case, we update the number of remaining iterations $it^h = it^h + 1$ and continue the process from the iteration point selected.

2. The DM is able to speed up or slow down the search process at any time by expressing a new value for the remaining number of iterations $it^h$.

3. At any iteration $h$, the intermediate points shown to the DM represent the set of solutions that can be reached from $\mathbf{z}^h$. However, the extreme solutions of the reachable region $P^h$ may not be represented by these points. This can be observed in Figure 3. We refer to the solutions which independently minimize each objective function over $P^h$ as *extreme solutions of $P^h$*. In practice, showing intermediate points associated to the extreme solutions at each iteration could be useful for the DM. They would allow her/him to gain information about the minimum value that each objective function could take from the current iteration point, and also about the price to be paid in the rest of the objectives in order to reach this minimum value. Because of that, intermediate points associated to the extreme solutions are stored in an archive set at each iteration. But, in order to avoid
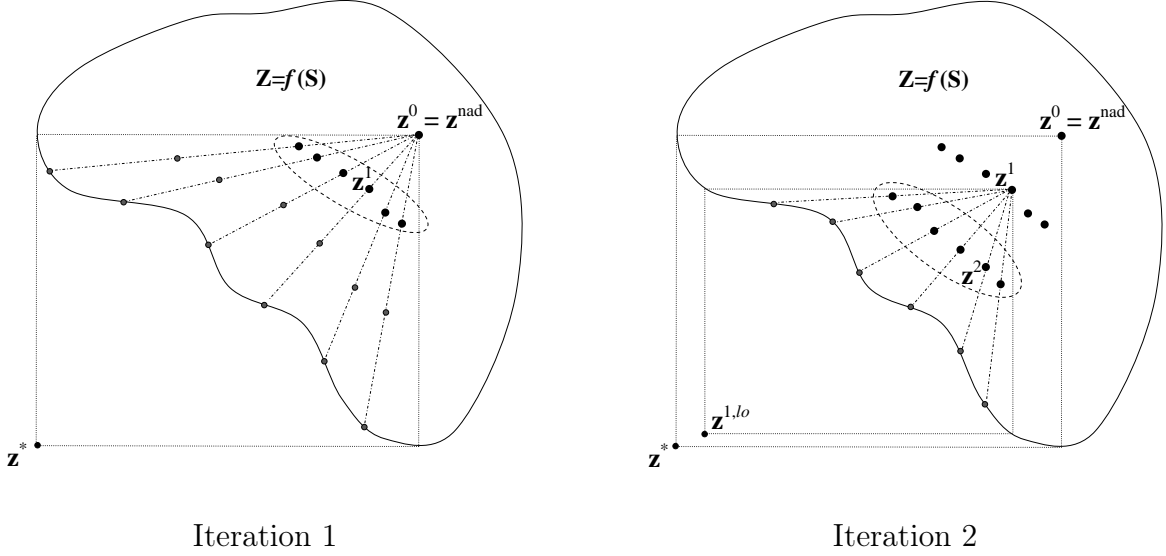
12

Figure 3: Graphical idea of the interactive decision making stage of E-NAUTILUS with two iterations. Starting from the nadir point, in the first iteration, the desired number of intermediate points is given to the DM. In the second iteration, from the iteration point selected (labelled as $\mathbf{z}^1$), the subset of reachable solutions at this iteration is reduced and, as it can be seen, new intermediate points are shown to the DM that represent it. This process continues until completing the number of iterations desired.

overwhelming the DM with too much information, these intermediate points are shown to the DM just in case (s)he explicitly asks for revising them at some iteration.

4. In Step 3 of Algorithm 1, during the selection of the most representative solutions in $P^h$, it may happen that the number of solutions in $P^h$ is less than $N_S$. This may be the case, for example, when the number of solutions in $P$ is not big enough and the DM is at the last iterations, when the proximity to $P$ is high. In this situation, there are at least two options: (a) the DM can be shown the set of intermediate points that can be obtained with the solutions currently available in $P^h$, and in this case the DM must be told that the required number of points is not available, or (b) additional solutions can be generated within the reachable part $P^h$ in order to get $N_S$ intermediate points as the DM wishes to see. For option (b), we propose to use a Pareto fill module in order to generate more nondominated solutions in the reachable part. This Pareto fill module is an auxiliary module and must be considered as an add-on to E-NAUTILUS. Hence, the module is not included in Algorithm 1.

In Figure 4, we enumerate the three steps involved in the Pareto fill module. In the first step, an interpolation method like PAINT (9) is used to create an approximation of the Pareto optimal front based on the solutions in $P$. In the second step, $\kappa$ uniformly distributed points are generated on a $k$-simplex (22) whose vertices ($\mathbf{u}^k$) are the bounds of the reachable part $P^h$ within which additional solutions are needed. The PAINT approximation is used in the third step to create a computationally inexpensive surrogate of the original problem (1). This surrogate problem is scalarized using the achievement scalarizing function proposed in (19). Next, the scalarized problem is solved several times by choosing one of the points generated on the $k$-simplex as a reference point at a time. The resulting solutions will lie on the PAINT approximation and they constitute a
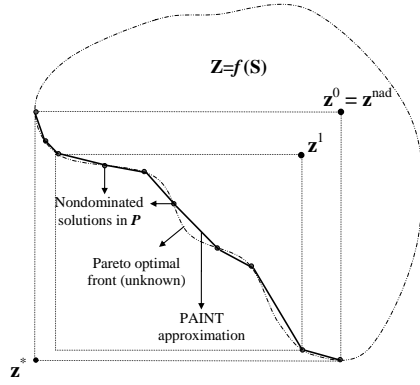
First step
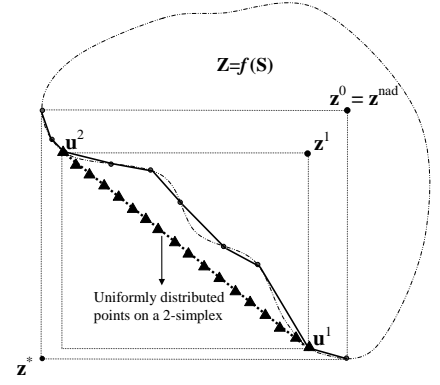


Second step
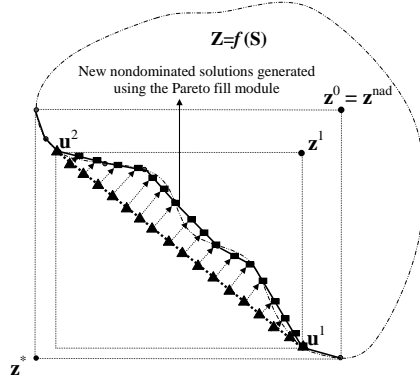


Third step

Figure 4: Graphical idea of the Pareto fill module. Firstly, an approximation of the Pareto optimal front is generated using an interpolation method like PAINT (9). Secondly, uniformly distributed points on a $k$-simplex are generated and, thirdly, these points are projected onto the PAINT approximation of the Pareto optimal front using the achievement sacalarizing function proposed in (19).

new set $P^h$, from which the Step 3 of Algorithm 1 can continue. Note, however, that the feasibility of the new solutions generated with the Pareto fill module cannot be guaranteed and, thus, the post-processing stage is strongly recommended in this case, since it finally guarantees the Pareto optimality of the final solution chosen by the DM. But if the projection in the post-processing stage is considered to be too time consuming, another option to guarantee at least the feasibility of the final solution is to find the solution in $P$ which has the minimum Euclidean distance from the point $\mathbf{z}^h$ finally chosen by the DM.

# 4  Numerical Example

In this section, we illustrate the performance of E-NAUTILUS in the real-life multiobjective optimization problem presented in (23), which is associated to the efficiency improvement of the auxiliary services of thermal power plants. This problem deals with finding the most convenient improvement strategies that can be carried out in the auxiliary services in order to maximize the energy saving achieved (denoted by $f_1$, in MWh), minimize the economic investment required

(denoted by $f_2$, in million €) and maximize the Internal Rate of Return (IRR) of the investment (denoted by $f_3$, in %). The improvement strategies considered are: (a) the replacement of the current electric motors by more efficient ones, (b) the installation of variable speed drives, and (c) compensation for reactive power. There are binary decision variables indicating whether the strategies (a) and (b) are implemented or not in the elements involved, as well as continuous decision variables containing the amount of reactive power to be compensated for in strategy (c) on each component of the auxiliary services. The case study considered is based on the auxiliary services of a 1,100 MW power plant, and the resulting multiobjective optimization problem has 13 continuous decision variables and 20 binary decision variables.

In this problem, objectives functions $f_1$ and $f_3$ must be maximized while $f_2$ must be minimized. In order to follow the notation of the paper, the original problem has been transformed into a minimization problem, where the objectives to be minimized are $-f_1$, $f_2$ and $-f_3$. However, in real practical problems, the objective values should be presented to the DM in the correct form (maximization/minimization) in order to ease the understanding of the values.

It should be said that we initially asked the DM for the importance he gave to each objective function. He indicated that his most important objective was the investment cost ($f_2$), given that he had to adjust to a budget of 2 million €, approximately. After that objective, the energy saving objective function ($f_1$) was the second most important objective for him and, finally, the IRR objective ($f_3$). The DM also said that, with E-NAUTILUS, he expected to find a solution with a satisfactory trade-off among the investment and the energy saving objectives. Note that this information is not needed by E-NAUTILUS, but it is very useful to have an initial understanding of the preferences and expectations of the DM.

In what follows, we describe the solution process using E-NAUTILUS. First, in the pre-processing stage, we started by finding a set of nondominated solutions $P$ which approximates the Pareto optimal front of the problem. The objective functions are discontinuous and nonconvex, even some of them are modelled using a black-box simulator due to the complex engineering formulas behind the model. Because of that, and the presence of binary and continuous variables, the Pareto optimal front was approximated using the evolutionary multiobjective optimization algorithms NSGA-II (24) and MOEA/D (25) (further details about the parameter setting used are available upon request). Since this is a real multiobjective optimization problem whose Pareto optimal front is unknown, we ran these two algorithms with a high population size (2000 individuals) and a large number of generations (650 generations) in order to be able to rely on the results obtained. The set $P$ was formed by the nondominated solutions of the final populations of both algorithms. As a result, $P$ had a total of 2218 solutions. The ideal and the nadir point were approximated from $P$ to get $\mathbf{z}^\star = (-47526.37, 0.05, -100.00)$ and $\mathbf{z}^{\mathrm{nad}} = (-408.49, 9.28, -22.13)$. From an overall perspective, after analysing the solutions obtained, we concluded that the highest values of the IRR, which were near 100%, were reached by solutions where the energy savings and the investment costs were not very high in comparison to the rest of the solutions. In contrast, solutions with higher energy savings and costs had lower IRR values, although these IRR values were still economically very profitable taking into account that the lowest IRR value attained was 22.13%.

Regarding the interactive decision making stage, two experiments are described hereafter. The first experiment carried out is described in Subsection 4.1 and shows the solution process followed when $P$ was formed by the 2218 solutions available. Although, in this example, we had a dense set of solutions approximating the Pareto optimal front, this may not be a common situation, specially in computationally complex multiobjective optimization problems. In these cases, the number of solutions in $P$ may be much lower in order to save in compuation cost. In

these situations, the Pareto fill module described in Section 3 may be internally needed during the solution process of E-NAUTILUS to generate extra solutions within the reachable part $P^h$, at some iteration $h$. In order to illustrate this, we have carried out a second experiment, described in Subsection 4.2, in which the size of $P$ was reduced by selecting the 100 most representative solutions available as the set $P$.

## 4.1  Experiment 1

As previously indicated, in this experiment the set $P$ consisted of 2118 nondominated solutions. After showing the ideal and the nadir points to the DM, he decided to take five iterations and wanted to see six points at each iteration, i.e., $N_I = 5$ and $N_S = 6$. Table 1 of B contains the information shown to the DM along the whole solution process. The first column of this table shows the iteration number $h$ and the second one the intermediate points given to the DM at each iteration. The *'Bounds'* column contains three subcolumns with the lower objective function bounds for each intermediate point (that is, the components of the vector $\mathbf{f}^{h,i,lo}$) and the last column *'Closeness'* indicates the value of $d^{h,i}$ for each point. In what follows, we describe the iterative process followed by the DM.

**Iteration 1** Based on the information provided, the DM observed that, from all the intermediate points of iteration 1, the lowest values that could be reached for the three objectives (their lower bounds) were still very close to their ideal values. Since he initially gave more importance to the investment cost than to the other two objectives, he decided to select the point with the lowest value for $f_2$ and chose the point $\mathbf{z}^{1,3}$. He noticed that this point had the worst value for $f_1$, but he did not mind having such a value since he was satisfied with the lowest bound that could be achieved for $f_1$ from $\mathbf{z}^{1,3}$. Then, we set $\mathbf{z}^1 = \mathbf{z}^{1,3}$.

**Iteration 2** At this iteration, apart from analysing the information given for $f_2$, the DM also paid special attention to the energy saving in $f_1$. He realized that the values that could be achieved by $f_2$ from all the intermediate points were lower than 2 million €, so he thought it would be worthy to improve $f_1$ rather than the $f_2$ value. From the intermediate points obtained, he preferred $\mathbf{z}^{2,2}$ because the lower bounds that could be reached for both objectives, $f_1$ and $f_2$, at this point were satisfactory enough. Then, we set $\mathbf{z}^2 = \mathbf{z}^{2,2}$.

**Iteration 3** Given that the lower bounds that $f_2$ could achieve from any of the intermediate points were very similar, and $f_3$ could also reach the same lower bound from all of them, the DM mainly concentrated on $f_1$ in order to select the most profitable point. According to that, he looked for the point with the best energy saving value and from which the best values for $f_1$ could be obtained. Then, he selected $\mathbf{z}^{3,3}$ as the next iteration point $\mathbf{z}^3$.

**Iteration 4** It can be seen that, at this iteration, the values for $f_2$ at all the intermediate points were greatly reduced in comparison to those of the previous iteration, and, at the same time, the DM could still find solutions which did not require to spend more than 2 million €. He found points $\mathbf{z}^{4,4}$, $\mathbf{z}^{4,5}$ and $\mathbf{z}^{4,6}$ interesting because, from all of them, the investment could be reduced down to 1.87 million €, and the IRR could be up to 61.47%. From these points, he initially selected the point $\mathbf{z}^{4,6}$ since it was the point with the best value for $f_1$ and from which the best lower bound for $f_1$ could be reached. However, at the next iteration, he was not satisfied with any of the intermediate points obtained because only one of them needed an investment below 2 million €. Given that he would

have chosen the final solution at that iteration (since it would have been the last one), he decided to take a step backwards. Then, among the three points he liked most at the fourth iteration, he looked for the point with the second best value for $f_1$. Finally, he selected $\mathbf{z}^{4,4}$ as the iteration point $\mathbf{z}^4$ in order to see if it was possible to obtain better results.

**Iteration 5** At the last iteration, we obtained solutions requiring to invest both more or less than 2 million €. Solutions $\mathbf{z}^{5,3}$ and $\mathbf{z}^{5,5}$ did not need more than 2 million €, but the DM also analysed the trade-offs among $f_1$ and $f_2$ in the solutions with an investment higher than 2 million €. Finally, the DM was more interested in the solution $\mathbf{z}^{5,4}$ than in $\mathbf{z}^{5,3}$ and $\mathbf{z}^{5,5}$. Although this solution required an investment over the initial budget, it saved more energy than $\mathbf{z}^{5,3}$ and $\mathbf{z}^{5,5}$ and the DM considered that the sacrifice to be paid in the investment for obtaining a better value for $f_1$ was worthy enough. Besides, among the solutions requiring more than 2 million €, $\mathbf{z}^{5,4}$ was the solution with the best IRR value ($f_3$), which also encouraged him to select this solution as the final one. Then, we set $\mathbf{z}_{pref}$ $= \mathbf{z}^5 = \mathbf{z}^{5,4}$.

In this experiment, the post-processing stage was not carried out since the computational effort required was regarded too high. Although we cannot assure that the solution $\mathbf{z}_{pref}$ is Pareto optimal, it belongs to $P$, which has been obtained by combining over 2000 individuals obtained by NSGA-II and MOEA/D, respectively, after 650 generations and by selecting the nondominated solutions among them.

A possible way to ease the interaction with the DM in E-NAUTILUS is to use a value path as the one showed in Figure 5 for graphically visualizing the solution process. In this value path, we illustrate the first and the second iterations of experiment 1. It can be seen that the DM can esily compare the objective values reached by all intermediate points at each iteration before selecting her/his most preferred point. Besides, at the second iteration, we illustrate the objective values which can be reached from the first iteration point $\mathbf{z}^1$ by highlighting with thick lines the part of each bar which corresponds to the reachable range for each objective function. The DM can also have an idea of how close the objectve objective values at the current iteration point are from their ideal values. Overall, what makes using this value path specially suitable for illustrating the E-NAUTILUS method is that the DM can graphically visualize the improvement of each objective along the whole solution process and the progression from the nadir point towards the final solution in $P$.

## 4.2   Experiment 2

Here, the set $P$ consisted of the 100 most representative solutions of the 2118 solutions initially generated. In this case, the ideal and the nadir points approximated from this set $P$ were $\mathbf{z}^\star = (-47, 504.84, 0.05, -99.02)$ and $\mathbf{z}^{\text{nad}} = (-413.47, 9.27, -22.16)$. Once the DM was shown the ideal and the nadir points, he indicated that he desired to see five solutions at each iteration and that he wanted to take five iterations, so we set $N_S = 5$ and $N_I = 5$. All the information provided to the DM in this second experiment can be seen in Table 2 in B.

**Iteration 1** Initially, the reachable values for $f_3$ and its lower bounds were very low in all the intermediate points obtained so, at this iteration, the DM gave more importance to $f_1$ and $f_2$. He decided to choose the point $\mathbf{z}^{1,3}$ given that it reached the second best value for $f_1$ and the investment needed at this point ($f_2$) was slightly lower than that of the
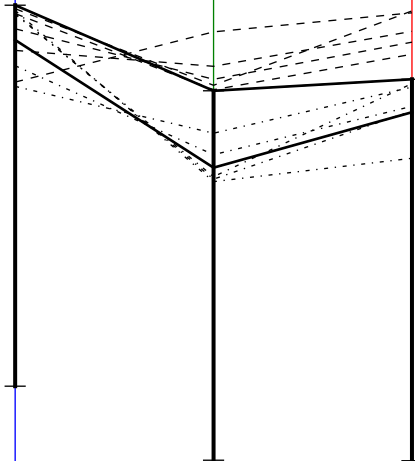
17

Figure 5: Two sets of value paths corresponding to the intermediate points shown to the DM at the first two iterations of experiment 1. From the solution selected in the first iteration, $\mathbf{z}^1$, the subset of reachable values for $f_1$, $f_2$ and $f_3$ is reduced, which has been highlighted with thick lines on each bar. Using a value path in this way, the DM can easily see at each iteration the improvement of each objective function and the shrinking ranges of the objective function values that are still reachable.

point with the lowest value for $f_1$. Besides, from $\mathbf{z}^{1,3}$, $f_1$ could reach very good values near its ideal value, and the reachable values for $f_2$ did not exceed the maximum amount of money he was willing to invest. Then, we set $\mathbf{z}^1 = \mathbf{z}^{1,3}$.

**Iteration 2** As $f_1$ and $f_3$ could still reach acceptable values for the DM in all the intermediate points found at this iteration, the DM was more interested in directing the search towards solutions which require the lowest investments ($f_2$). The point with the lowest $f_2$ value was $\mathbf{z}^{2,1}$, and it must be noted that similar lower bounds for $f_2$ and $f_3$ were obtained by the points $\mathbf{z}^{2,1}$ and $\mathbf{z}^{2,3}$. However, the reachable value for $f_1$ from $\mathbf{z}^{2,1}$ was the lowest one. According to that, the DM realized that more profitable solutions might be found from $\mathbf{z}^{2,1}$ than from $\mathbf{z}^{2,3}$, so he selected it and we set $\mathbf{z}^2 = \mathbf{z}^{2,1}$.

**Iteration 3** From the information shown at iteration 3, the DM observed that the $f_2$ values were significantly reduced at all the intermediate points when compared to $\mathbf{z}^2$ and he liked this improvement a lot. From them, he selected as the next iteration point $\mathbf{z}^3$ the point $\mathbf{z}^{3,1}$, since this was the point from which the best $f_2$ and $f_3$ values could be obtained, and, at the same time, he thought that the reachable values for $f_1$ were still very good. Thus, we set $\mathbf{z}^3 = \mathbf{z}^{3,1}$.

**Iteration 4** At the fourth iteration, the DM put more attention on the first objective than in the other two because he was satisfied with the values obtained for $f_2$ and $f_3$ in all the intermediate points. He observed that a significant improvement in $f_1$ was obtained by all the points. The point he liked most was $\mathbf{z}^{4,5}$ because this point was the one with the second best value for $f_1$, and the DM realised that, from it, he could reach the lowest value for $f_1$ and the second lowest value for $f_2$. Then, we set $\mathbf{z}^4 = \mathbf{z}^{4,5}$. After that, the DM

18

noticed that the closeness to $P$ was almost 80% and only one iteration remained ($it^h = 1$), which meant that he would choose his last preferred solution at the next iteration. But he wished to explore the reachable part of $P$ from $\mathbf{z}^4$ more carefully, so he decided to slow down the search and to take two iterations more (internally, we updated $it^h = 3$).

**Iteration 5** From the intermediate points shown, the point the DM liked most was $\mathbf{z}^{5,2}$ and we set $\mathbf{z}^5 = \mathbf{z}^{5,2}$. There were several points with the same lower bound for $f_2$, but the DM was most interested in this point because it achieved the best value for the energy saving ($f_1$) among all of them and, at the same time, the values that could still be reached for $f_1$ from it were among the best ones. Regarding $f_3$, the DM was satisfied with the values achieved and lower bounds obtained for this objective by all the intermediate points, so he did not concentrate on that objective too much. At this iteration, the DM was 84.47% close to the set $P$ and he was quite satisfied with the lower bound values that the objective functions could reach from $\mathbf{z}^5$. Then, he decided to increase the number of points to be obtained at the next two iterations to have more information about the solutions that could be reached from $\mathbf{z}^5$. He desired to see ten points, so we updated $N_S = 10$.

**Iteration 6** Since only one iteration remained, he concentrated on revising the lower objective function bounds that could be obtained from each one of the 10 intermediate points obtained. Analysing the lower bounds obtained for $f_1$, he could distinguish two kinds of intermediate points. On the one hand, there was a first group of intermediate points from which $f_1$ could reach values around -26000.00 (points $\mathbf{z}^{6,1}$, $\mathbf{z}^{6,3}$, $\mathbf{z}^{6,4}$, $\mathbf{z}^{6,5}$, $\mathbf{z}^{6,8}$ and $\mathbf{z}^{6,10}$). On the other hand, the second group was formed by the intermediate points $\mathbf{z}^{6,2}$, $\mathbf{z}^{6,6}$, $\mathbf{z}^{6,7}$ and $\mathbf{z}^{6,9}$, from which better lower bounds for $f_1$ were obtained (around -32000.00) in comparison to the points in the first group. However, the points in the second group had obtained lower bounds for $f_2$ which were worse than those of the points in the first group and, regarding $f_3$, better values could be reached from the points in the second group than from the points in the first one. As a result, from the intermediate points in the first group, the DM could reach a final solution which saved less energy, required less money but reached a worse IRR value than the solution that could reach from the intermediate points in the second group. Based on this, and given that his initial budget was around 2 million €, the DM decided to select a point in the first group, although higher energy savings could be obtained from the points in the other group at the expense of a higher investment. Then, among the points in the first group, he selected $\mathbf{z}^{6,6}$ because it was the point from which the best energy savings ($f_1$) could be reached ($\mathbf{z}^6 = \mathbf{z}^{6,6}$).

**Iteration 7** It should me mentioned that, at this iteration, the Pareto fill module described in Section 3 was internally needed. The iteration point $\mathbf{z}^6$ was very close to the set $P$ (at 94.34% of the total distance) and, internally, there were not enough solutions in the subset $P^7$ to find the 10 representative solutions required to calculate the 10 intermediate points. This module generates a set of extra solutions within the reachable region $P^7$ and, since this is the last iteration, the intermediate points generated were nondominated solutions belonging to this set. Based on the previous experiment, and although none of the points obtained required an investment below 2 million €, the DM did not want to go backwards since he was satisfied with the trade-offs among $f_1$ and $f_2$. He finally selected the point $\mathbf{z}^{7,3}$ as the his most preferred solution. It had the lowest value for $f_2$ and, at the same time, it reached the best value for $f_3$, which meant that this was the most profitable solution. Consequently, we set $\mathbf{z}_{pref} = \mathbf{z}^7 = \mathbf{z}^{7,3}$.
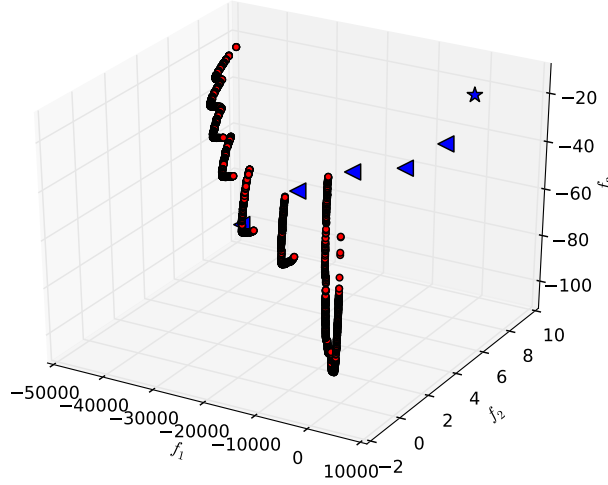
Figure 6: Trajectory of the iteration points selected by the DM in experiment 1 (the Pareto fill module is not used). The star represents the nadir point of the problem, the solutions approximating the Pareto optimal front are the dark encircled points and the iteration points selected by the DM at different iterations of E-NAUTILUS are represented by triangles.

In this experiment, given that the Pareto fill module was needed at the last iteration, the post-processing stage is strongly recommended in order to assure the Pareto optimality of $\mathbf{z}_{pref}$. In our case, the post-processing stage was carried out by selecting the closest solution of the set $P$ considered in this experiment to $\mathbf{z}_{pref}$ with respect to the Euclidean distance, which was the solution $(-25368.00, 2.1438, -58.58)$.

In Figures 6 and 7, the star indicates the nadir point of the problem, the dark encircled points represent the solutions in $P$ which approximate the Pareto optimal front and the triangles are the iteration points selected by the DM at different iterations of E-NAUTILUS, in experiments 1 and 2, respectively. Since in experiment 2 (Figure 7) the solution $\mathbf{z}_{pref}$ and the final solution are different, we have represented the final solution with a large (yellow) circle. The figure clearly shows that the DM made consistent progress by choosing a solution better than the previous solution and finally obtained a solution in the same region than in experiment 1. Additionally, using the Pareto fill module in experiment 2 did not drastically change the final solution preferable to the DM as compared to experiment 1.

## 5   Conclusions

In this paper, we have proposed enhancements to the interactive method NAUTILUS and called it the E-NAUTILUS method. In principle, E-NAUTILUS is aimed at avoiding the undesired anchoring effects which may take place in interactive solution processes. It progressively proceeds to the most preferred Pareto optimal solution from the nadir point, taking one step closer to the Pareto optimal set at each iteration, and effectively handles computationally expensive multiobjective optimization problems. What is important is that the DM does not face the computational cost during the solution process, even if the problem is computationally expensive.

E-NAUTILUS is divided into three stages. In the first stage, called the pre-processing stage, a set of well-spread nondominated solutions approximating the entire Pareto optimal front is
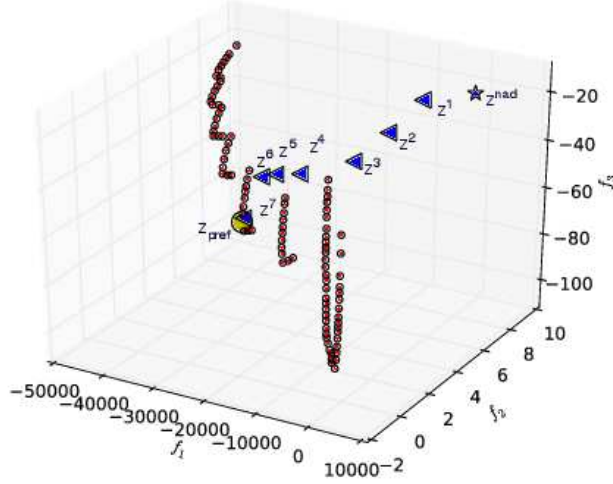
Figure 7: Trajectory of the iteration points selected by the DM in experiment 2 (the Pareto fill module is used). The star represents the nadir point of the problem, the solutions approximating the Pareto optimal front are the dark encircled points, the iteration points selected by the DM at different iterations of E-NAUTILUS are represented by triangles and the large (yellow) circle is the final solution.

generated. Additionally, if a well-spread set of solutions cannot be generated, approximation techniques such as the PAINT method can be used to generate more solutions without solving the original problem. The next stage, called the interactive decision making stage, constitutes the core of E-NAUTILUS and is the stage where the DM interacts with the solution process. The DM progressively proceeds from the nadir point towards the most preferred solution in the set of nondominated solutions generated in the previous stage. At every iteration, the DM is given a set of intermediate points, from which (s)he must select the next iteration point. It is guaranteed that all the intermediate points shown at each iteration to the DM are mutually nondominated, as well as that all of them dominate the previous iteration point chosen by the DM. In this way, the DM always obtains an improvement in all objective functions at every iteration, thereby keeping her/him always motivated and focused. The DM is also provided with additional information regarding the kind of solutions that may be reachable from each iteration point before making a decision. At the last iteration, it is assured that the DM selects one solution which belongs to the set of nondominated solutions which were initially generated. However, given that this set just constitutes an approximation of the Pareto optimal front, it is not assured that the last solution selected by the DM is Pareto optimal. Because of this, we need the last stage, called the post-processing stage, in which an achievement scalarizing function is formulated and solved to project the final solution to be Pareto optimal.

In addition, some flexibilities for the interactive decision making stage of E-NAUTILUS have been proposed, such as changing the number of solutions to be seen or the number of iterations left at any time, and the possibility of going backwards at any iteration. We have also suggested an add-on module to be used in case the number of solutions that can still be reached is not big enough to show the DM the number of intermediate points (s)he desires to

see. In this case, this module generates additional nondominated solutions in the reachable part by means of an interpolation method.

We have illustrated the E-NAUTILUS method using a real-life multiobjective optimization problem concerning the improvement strategies of the auxiliary services of thermal power plants. Here the DM progressively improved the values of the objective functions at each generation according to his preference information and he finally reached a final solution which responded to his initial expectations. For the sake of demonstration a second experiment was performed using a reduced set of nondominated solutions in the pre-processing stage. In the second experiment, an approximation of nondominated solutions using the PAINT method was activated at the last iteration due to an insufficient number of solutions available. However, the preferred solution that was chosen by the DM was not very different from the one in experiment 1. Thus, the additional flexibility added to the E-NAUTILUS is justified.

Future research directions include a further study of graphical illustrations that enable showing the DM the information in an understandable and intuitive way, and the development of a user-friendly interface to interact with the DM. The role of a user-friendly interface is essential so that DMs can make the most of the properties of E-NAUTILUS.

# Acknowledgements

# A    Theorems and Proofs

From now on, for any $h$, let us consider $\beta^h = \frac{it^h - 1}{it^h}$. Given that $it^h$ is an integer value between 1 and $N_I$, we have $\beta^h \in [0, 1)$ for any $h$. Then, using (2), $\mathbf{z}^{h,i}$ can be expressed in the following way for any $i = 1, \ldots, N_S$:

$$\mathbf{z}^{h,i} = \beta^h \mathbf{z}^{h-1} + (1 - \beta^h)\bar{\mathbf{z}}^{h,i}. \tag{5}$$

**Theorem 1.** *At any iteration $h$, for any $r_1, r_2 \in \{1, \ldots, N_S\}$ solutions $\mathbf{z}^{h,r_1}$ and $\mathbf{z}^{h,r_2}$ do not dominate each other.*

*Proof.* Let us consider any two $r_1, r_2 \in \{1, \ldots, N_S\}$.

1. If $it^h = 1$, from (5), we have $\mathbf{z}^{h,r_1} = \bar{\mathbf{z}}^{h,r_1}$ and $\mathbf{z}^{h,r_2} = \bar{\mathbf{z}}^{h,r_2}$. Since $\bar{\mathbf{z}}^{h,r_1}, \bar{\mathbf{z}}^{h,r_2} \in P$, they do not dominate each other and this completes the proof.

2. If $it^h > 1$, let us assume that $\mathbf{z}^{h,r_1}$ dominates $\mathbf{z}^{h,r_2}$. This means that $z_j^{h,r_1} \leq z_j^{h,r_2}$ for every $j = 1, \ldots, k$ and there exists $s$ such that $z_s^{h,r_1} < z_s^{h,r_2}$. Using (5), this implies that

$$\beta^h z_j^{h-1} + (1 - \beta^h)\bar{z}_j^{h,r_1} \leq \beta^h z_j^{h-1} + (1 - \beta^h)\bar{z}_j^{h,r_2} \text{ for every } j = 1, \ldots, k \text{ and}$$

$$\beta^h z_s^{h-1} + (1 - \beta^h)\bar{z}_s^{h,r_1} < \beta^h z_s^{h-1} + (1 - \beta^h)\bar{z}_s^{h,r_2} \Rightarrow$$

22

$(1 - \beta^h)\bar{z}_j^{h,r_1} \leq (1 - \beta^h)\bar{z}_j^{h,r_2}$ for every $i = 1, \ldots, k$ and $(1 - \beta^h)\bar{z}_s^{h,r_1} < (1 - \beta^h)\bar{z}_s^{h,r_2}$.

Given that $\beta^h < 1$, we can simplify as follows:

$$\bar{z}_j^{h,r_1} \leq \bar{z}_j^{h,r_2} \text{ for every } i = 1, \ldots, k \text{ and } \bar{z}_s^{h,r_1} < \bar{z}_s^{h,r_2}.$$

This means that $\bar{\mathbf{z}}^{h,r_1}$ dominates $\bar{\mathbf{z}}^{h,r_2}$, but this is not possible because $\bar{\mathbf{z}}^{h,r_1}, \bar{\mathbf{z}}^{h,r_2} \in P$ and they do not dominate each other.

$\square$

**Theorem 2.** *1. At any iteration $h$ with $it^h \neq 1$, the objective vector $\mathbf{z}^{h,i}$ is an achievable point dominated by $\bar{\mathbf{z}}^{h,i}$, for any $i = 1, \ldots, N_S$.*

*2. At any iteration $h$, $\mathbf{z}^h$ dominates $\mathbf{z}^{h-1}$.*

*Proof.* 1. Since we just consider values of $h$ with $it^h \neq 1$, we can assume that $\beta^h \neq 0$ for every $h$. Then, $\mathbf{z}^{h,i} \neq \bar{\mathbf{z}}^{h,i}$ for every $i = 1, \ldots, N_S$.

Let us prove the theorem by induction. First, let us show that $\mathbf{z}^{1,i}$ is dominated by $\bar{\mathbf{z}}^{1,i}$, for every $i = 1, \ldots, N_S$. If we set $h = 1$ in (5), we have

$$\mathbf{z}^{1,i} = \beta^1 \mathbf{z}^0 + (1 - \beta^1)\bar{\mathbf{z}}^{1,i}. \tag{6}$$

Given that $\mathbf{z}^0 = \mathbf{z}^{\text{nad}}$ and $\bar{\mathbf{z}}^{1,i} \in P^h = P$ (the approximation of the Pareto optimal front), we can assume that $\bar{\mathbf{z}}^{1,i}$ dominates $\mathbf{z}^0$, so $\bar{z}_j^{1,i} \leq z_j^0$ for every $j = 1, \ldots, k$ and $\bar{z}_s^{1,i} < z_s^0$ for at least one index $s$. Then, from (6), we have

$$z_j^{1,i} = \beta^1 z_j^0 + (1 - \beta^1)\bar{z}_j^{1,i} \geq \beta^1 \bar{z}_j^{1,i} + (1 - \beta^1)\bar{z}_j^{1,i} = \bar{z}_j^{1,i} \text{ for every } j = 1, \ldots, k \text{ and}$$

$$z_s^{1,i} = \beta^1 z_s^0 + (1 - \beta^1)\bar{z}_s^{1,i} > \beta^1 \bar{z}_s^{1,i} + (1 - \beta^1)\bar{z}_s^{1,i} = \bar{z}_s^{1,i} \Rightarrow$$

$$z_j^{1,i} \geq \bar{z}_j^{1,i} \text{ for every } j = 1, \ldots, k \text{ and } z_s^{1,i} > \bar{z}_s^{1,i}.$$

Consequently, $\mathbf{z}^{1,i}$ is an achievable point dominated by $\bar{\mathbf{z}}^{1,i}$.

Let us assume that $\mathbf{z}^{h-1,i}$ is an achievable point dominated by $\bar{\mathbf{z}}^{h-1,i}$ for every $i = 1, \ldots, N_S$, and let us prove that $\mathbf{z}^{h,i}$ is an achievable point dominated by $\bar{\mathbf{z}}^{h,i}$ for every $i = 1, \ldots, N_S$.

Let us show that, for any $\mathbf{z} \in P^{h-1}$, there exists one index $s$ for which $z_s < z_s^{h-1}$, which will imply that all the solutions in $P^h$ dominate $\mathbf{z}^{h-1}$. On the one hand, $\mathbf{z}^{h-1}$ is the previous iteration point, so $\mathbf{z}^{h-1} = \mathbf{z}^{h-1,l}$ for one index $l = 1, \ldots, N_S$. Since we have supposed that $\mathbf{z}^{h-1,l}$ is dominated by $\bar{\mathbf{z}}^{h-1,l}$, then $\mathbf{z}^{h-1}$ is dominated by $\bar{\mathbf{z}}^{h-1,l}$. On the other hand, we have $z_j \leq z_j^{h-1}$ for all $j = 1, \ldots, k$ (as indicated in Step 8 of Algorithm 1, any $\mathbf{z} \in P^{h-1}$ satisfies $f_j^{h-1,l,lo} \leq z_j \leq z_j^{h-1}$ for all $j = 1, \ldots, k$). Consequently, if we assume that there does not exist any $s$ for which $z_s < z_s^{h-1}$, we have $\mathbf{z} = \mathbf{z}^{h-1}$ for every $\mathbf{z} \in P^{h-1}$. Then, $\mathbf{z}^{h-1}$ belongs to $P$ and thus, it is not dominated by any other solution in $P$. But this contradicts the fact that $\mathbf{z}^{h-1}$ is dominated by $\bar{\mathbf{z}}^{h-1,l} \in P$. With this, we have shown that all the solutions in $P^h$ dominate $\mathbf{z}^{h-1}$.

Given that $\bar{\mathbf{z}}^{h,i} \in P^h$ for every $i = 1, \ldots, N_S$, $\bar{\mathbf{z}}^{h,i}$ dominates $\mathbf{z}^{h-1}$. Then, $\bar{z}_j^{h,i} \leq z_j^{h-1}$ for every $j = 1, \ldots, k$ and $\bar{z}_s^{h,i} < z_s^{h-1}$ for at least one index $s$. Consequently, using (5), we have

$$z_j^{h,i} = \beta^h z_j^{h-1} + (1 - \beta^h)\bar{z}_j^{h,i} \geq \beta^h \bar{z}_j^{h,i} + (1 - \beta^h)\bar{z}_j^{h,i} = \bar{z}_j^{h,i} \text{ for every } j = 1, \ldots, k \text{ and}$$

$$z_s^{h,i} = \beta^h z_s^{h-1} + (1 - \beta^h)\bar{z}_s^{h,i} > \beta^h \bar{z}_s^{h,i} + (1 - \beta^h)\bar{z}_s^{h,i} = \bar{z}_s^{h,i} \Rightarrow$$

$$z_j^{h,i} \geq \bar{z}_j^{h,i} \text{ for every } j = 1, \ldots, k \text{ and } z_s^{h,i} > \bar{z}_s^{h,i}.$$

This implies that, for every $i = 1, \ldots, N_S$, $\mathbf{z}^{h,i}$ is an achievable point dominated by $\bar{\mathbf{z}}^{h,i}$ and the proof is complete.

2. At any iteration $h$, we know that $\mathbf{z}^h = \mathbf{z}^{h,r}$, for some index $r = 1, \ldots, N_S$. We must distinguish two cases:

- If $it^h = 1$, then $\beta^h = 0$ and we have $\mathbf{z}^{h,i} = \bar{\mathbf{z}}^{h,i}$ for every $i = 1, \ldots, N_S$. Since $\bar{\mathbf{z}}^{h,i} \in P^h$ and the solutions in $P^h$ dominate $\mathbf{z}^{h-1}$ (see Step 8 of Algorithm 1), $\bar{\mathbf{z}}^{h,i}$ dominates $\mathbf{z}^{h-1}$ for every $i = 1, \ldots, N_S$. In particular, $\mathbf{z}^h = \mathbf{z}^{h,r}$ dominates $\mathbf{z}^{h-1}$ and the proof is complete in this case.

- If $it^h \neq 1$, then $\beta^h \neq 0$. Theorem 2 assures that $\mathbf{z}^{h,r}$ is dominated by $\bar{\mathbf{z}}^{h,r}$, so $\bar{z}_j^{h,r} \leq z_j^{h,r} = z_j^h$ for every $j = 1, \ldots k$ and $\bar{z}_s^{h,r} < z_s^{h,r} = z_s^h$ for one index $s$. Then, using (5), we have

$$z_j^h = z_j^{h,r} = \beta^h z_j^{h-1} + (1 - \beta^h)\bar{z}_j^{h,r} \leq \beta^h z_j^{h-1} + (1 - \beta^h)z_j^h \text{ for every } j = 1, \ldots, k$$

$$\text{and } z_s^h = z_s^{h,r} = \beta^h z_s^{h-1} + (1 - \beta^h)\bar{z}_s^{h,r} < \beta^h z_s^{h-1} + (1 - \beta^h)z_s^h \Rightarrow$$

$$z_j^h - (1 - \beta^h)z_j^h \leq \beta^h z_j^{h-1} \text{ for every } j = 1, \ldots, k \text{ and } z_s^h - (1 - \beta^h)z_s^h < \beta^h z_s^{h-1} \Rightarrow$$

$$\beta^h z_j^h \leq \beta^h z_j^{h-1} \text{ for every } j = 1, \ldots, k \text{ and } \beta^h z_s^h < \beta^h z_s^{h-1}.$$

Given that $\beta^h \neq 0$, we can simplify and get

$$z_j^h \leq z_j^{h-1} \text{ for every } j = 1, \ldots, k \text{ and } z_s^h < z_s^{h-1},$$

what means that $\mathbf{z}^h$ dominates $\mathbf{z}^{h-1}$.

$\square$

# B    Tables of the Experiments

| Iteration number ($h$) | Intermediate points $(f_1, f_2, f_3)$ | Bounds $f_1^{h,lo}$ | $f_2^{h,lo}$ | $f_3^{h,lo}$ | Closeness (%) |
|---|---|---|---|---|---|
| 1 | $\mathbf{z}^{1,1} = (-1307.16,\ 7.57,\ -23.99)$ | -46531.03 | 0.07 | -100.00 | 20.00 |
| | $\mathbf{z}^{1,2} = (-1085.79,\ 7.48,\ -31.31)$ | -44268.72 | 0.06 | -100.00 | 20.00 |
| | $\mathbf{z}^{1,3} = (\mathbf{-949.27},\ \mathbf{7.46},\ \mathbf{-35.54})$ | **-39905.81** | **0.06** | **-100.00** | **20.00** |
| | $\mathbf{z}^{1,4} = (-5569.30,\ 7.95,\ -27.44)$ | -46892.30 | 0.96 | -65.73 | 20.00 |
| | $\mathbf{z}^{1,5} = (-3360.41,\ 7.69,\ -29.27)$ | -44558.82 | 0.19 | -87.87 | 20.00 |
| | $\mathbf{z}^{1,6} = (-8808.42,\ 8.64,\ -24.31)$ | -47423.13 | 0.96 | -65.73 | 20.00 |
| 2 | $\mathbf{z}^{2,1} = (-1764.56,\ 5.69,\ -40.87)$ | -34104.70 | 0.09 | -100.00 | 43.01 |
| | $\mathbf{z}^{2,2} = (\mathbf{-4503.98},\ \mathbf{5.92},\ \mathbf{-41.11})$ | **-34085.53** | **0.96** | **-65.73** | **43.48** |
| | $\mathbf{z}^{2,3} = (-9249.67,\ 6.61,\ -36.86)$ | -39565.26 | 0.96 | -65.73 | 41.68 |
| | $\mathbf{z}^{2,4} = (-1493.23,\ 5.64,\ -48.89)$ | -26178.51 | 0.07 | -100.00 | 40.03 |
| | $\mathbf{z}^{2,5} = (-1877.87,\ 5.74,\ -36.27)$ | -39676.58 | 0.09 | -100.00 | 43.90 |
| | $\mathbf{z}^{2,6} = (-7139.30,\ 6.18,\ -40.11)$ | -34208.04 | 0.96 | -65.73 | 42.81 |
| 3 | $\mathbf{z}^{3,1} = (-8221.39,\ 4.47,\ -43.79)$ | -33660.84 | 0.96 | -65.71 | 63.86 |
| | $\mathbf{z}^{3,2} = (-7424.58,\ 4.29,\ -48.76)$ | -26182.83 | 0.96 | -65.71 | 60.90 |
| | $\mathbf{z}^{3,3} = (\mathbf{-11872.92},\ \mathbf{4.93},\ \mathbf{-42.11})$ | **-33823.30** | **0.96** | **-65.71** | **63.54** |
| | $\mathbf{z}^{3,4} = (-11653.55,\ 4.79,\ -44.29)$ | -33430.78 | 0.96 | -65.71 | 62.36 |
| | $\mathbf{z}^{3,5} = (-11369.47,\ 4.64,\ -47.26)$ | -32919.19 | 0.96 | -65.71 | 60.76 |
| | $\mathbf{z}^{3,6} = (-14022.17,\ 5.11,\ -42.88)$ | -33803.39 | 1.07 | -65.71 | 60.76 |
| 4 | $\mathbf{z}^{4,1} = (-13746.05,\ 3.24,\ -45.80)$ | -26463.06 | 1.05 | -65.71 | 80.42 |
| | $\mathbf{z}^{4,2} = (-12569.34,\ 2.98,\ -53.08)$ | -25827.84 | 0.99 | -65.71 | 77.05 |
| | $\mathbf{z}^{4,3} = (-22420.28,\ 4.19,\ -44.40)$ | -33430.78 | 1.88 | -61.47 | 78.69 |
| | $\mathbf{z}^{4,4} = (\mathbf{-18788.82},\ \mathbf{3.64},\ \mathbf{-48.02})$ | **-26270.44** | **1.87** | **-61.47** | **79.02** |
| | $\mathbf{z}^{4,5} = (-17803.90,\ 3.44,\ -51.35)$ | -25943.97 | 1.87 | -61.47 | 77.77 |
| | $\mathbf{z}^{4,6} = (-19163.92,\ 3.87,\ -44.06)$ | -33430.78 | 1.87 | -61.47 | 80.78 |
| 5 | $\mathbf{z}^{5,1} = (-25562.47,\ 2.27,\ -55.61)$ | | | | |
| | $\mathbf{z}^{5,2} = (-25795.03,\ 2.38,\ -53.39)$ | | | | |
| | $\mathbf{z}^{5,3} = (-23058.31,\ 1.90, -60.07)$ | | | | |
| | $\mathbf{z}^{5,4} = (\mathbf{-25306.11},\ \mathbf{2.13},\ \mathbf{-58.88})$ | | | | |
| | $\mathbf{z}^{5,5} = (-24540.09,\ 1.99,\ -61.28)$ | | | | |
| | $\mathbf{z}^{5,6} = (-26029.09,\ 2.54,\ -50.30)$ | | | | |

Table 1: Experiment 1: Data provided to the DM and DM's choices in bold face.

| Iteration number ($h$) | Intermediate points $(f_1, f_2, f_3)$ | Bounds $f_1^{h,lo}$ | $f_2^{h,lo}$ | $f_3^{h,lo}$ | Closeness (%) |
|---|---|---|---|---|---|
| 1 | $\mathbf{z}^{1,1} = (-1210.39,\ 7.51,\ -26.56)$ | -45202.61 | 0.07 | -99.02 | 20.00 |
| | $\mathbf{z}^{1,2} = (-1061.24,\ 7.46,\ -33.84)$ | -39980.74 | 0.06 | -99.02 | 20.00 |
| | $\mathbf{z}^{1,3} = (\mathbf{-8243.83},\ \mathbf{8.44},\ \mathbf{-25.11})$ | **-47277.75** | **0.97** | **-65.64** | **20.00** |
| | $\mathbf{z}^{1,4} = (-5443.27,\ 7.87,\ -28.85)$ | -44523.93 | 0.97 | -65.64 | 20.00 |
| | $\mathbf{z}^{1,5} = (-9730.37,\ 8.99,\ -23.16)$ | -47463.28 | 0.97 | -65.64 | 20.00 |
| 2 | $\mathbf{z}^{2,1} = (\mathbf{-10003.36},\ \mathbf{6.67},\ \mathbf{-32.58})$ | **-43742.27** | **1.02** | **-65.64** | **37.02** |
| | $\mathbf{z}^{2,2} = (-17281.97,\ 8.02,\ -26.49)$ | -46997.96 | 1.89 | -61.28 | 39.54 |
| | $\mathbf{z}^{2,3} = (-12456.40,\ 6.85,\ -33.84)$ | -39980.74 | 1.02 | -65.64 | 37.38 |
| | $\mathbf{z}^{2,4} = (-13863.23,\ 7.14,\ -30.36)$ | -44396.37 | 1.09 | -65.64 | 39.50 |
| | $\mathbf{z}^{2,5} = (-15310.76,\ 7.48,\ -28.33)$ | -44654.26 | 1.44 | -61.28 | 40.34 |
| 3 | $\mathbf{z}^{3,1} = (\mathbf{-11860.17},\ \mathbf{4.95},\ \mathbf{-38.48})$ | **-38633.59** | **1.02** | **-65.64** | **58.53** |
| | $\mathbf{z}^{3,2} = (-11607.41,\ 4.84,\ -42.56)$ | -33776.49 | 1.02 | -65.64 | 56.04 |
| | $\mathbf{z}^{3,3} = (-16909.38,\ 5.53,\ -37.09)$ | -39179.87 | 1.89 | -61.28 | 58.88 |
| | $\mathbf{z}^{3,4} = (-19162.30,\ 6.00,\ -34.59)$ | -39980.74 | 1.89 | -61.28 | 58.24 |
| | $\mathbf{z}^{3,5} = (-15033.61,\ 5.14,\ -41.73)$ | -33776.49 | 1.24 | -61.28 | 56.68 |
| 4 | $\mathbf{z}^{4,1} = (-13716.98,\ 3.24,\ -44.39)$ | -26600.87 | 1.09 | -65.64 | 79.18 |
| | $\mathbf{z}^{4,2} = (-13337.85,\ 3.06,\ -50.50)$ | -25795.03 | 1.09 | -65.64 | 76.35 |
| | $\mathbf{z}^{4,3} = (-22818.33,\ 4.38,\ -40.79)$ | -33776.49 | 1.93 | -61.28 | 78.17 |
| | $\mathbf{z}^{4,4} = (-18477.14,\ 3.51,\ -49.26)$ | -25957.62 | 1.89 | -61.28 | 76.61 |
| | $\mathbf{z}^{4,5} = (\mathbf{-19102.12},\ \mathbf{3.85},\ \mathbf{-42.61})$ | **-33776.49** | **1.89** | **-61.28** | **79.62** |
| 5 | $\mathbf{z}^{5,1} = (-21255.57,\ 3.32,\ -46.94)$ | -26226.15 | 1.93 | -60.35 | 82.53 |
| | $\mathbf{z}^{5,2} = (\mathbf{-23278.79},\ \mathbf{3.67},\ \mathbf{-44.01})$ | **-33391.72** | **1.93** | **-60.35** | **84.47** |
| | $\mathbf{z}^{5,3} = (-21476.80,\ 3.45,\ -44.51)$ | -32924.97 | 1.93 | -60.35 | 85.67 |
| | $\mathbf{z}^{5,4} = (-21099.45,\ 3.26,\ -48.42)$ | -25957.62 | 1.93 | -60.35 | 80.75 |
| | $\mathbf{z}^{5,5} = (-23993.58,\ 3.84,\ -42.77)$ | -33776.49 | 2.07 | -60.04 | 85.01 |
| 6 | $\mathbf{z}^{6,1} = (-24420.63,\ 2.97,\ -49.81)$ | -25957.62 | 2.07 | -60.04 | 90.26 |
| | $\mathbf{z}^{6,2} = (-27000.11,\ 3.46,\ -45.06)$ | -32924.97 | 3.26 | -47.21 | 92.65 |
| | $\mathbf{z}^{6,3} = (-24618.20,\ 3.10,\ -47.22)$ | -26226.15 | 2.07 | -60.04 | 92.14 |
| | $\mathbf{z}^{6,4} = (-24186.45,\ 2.87,\ -52.03)$ | -25795.03 | 2.07 | -60.04 | 88.73 |
| | $\mathbf{z}^{6,5} = (-24811.43,\ 3.21,\ -45.37)$ | -26344.07 | 2.07 | -60.04 | 93.53 |
| | $\mathbf{z}^{6,6} = (\mathbf{-24939.83},\ \mathbf{3.29},\ \mathbf{-44.26})$ | **-30721.43** | **2.07** | **-60.04** | **94.39** |
| | $\mathbf{z}^{6,7} = (-27455.46,\ 3.49,\ -45.41)$ | -32924.97 | 3.26 | -47.21 | 92.07 |
| | $\mathbf{z}^{6,8} = (-24752.47,\ 3.17,\ -46.17)$ | -26344.07 | 2.07 | -60.04 | 92.92 |
| | $\mathbf{z}^{6,9} = (-28101.88,\ 3.54,\ -45.61)$ | -32924.97 | 3.26 | -47.21 | 91.45 |
| | $\mathbf{z}^{6,10} = (-24536.91,\ 3.03,\ -48.70)$ | -25957.62 | 2.07 | -60.04 | 91.05 |
| 7 | $\mathbf{z}^{7,1} = (-26100.00,\ 2.56,\ -50.10)$ | | | | |
| | $\mathbf{z}^{7,2} = (-25900.00,\ 2.48,\ -51.50)$ | | | | |
| | $\mathbf{z}^{7,3} = (\mathbf{-25300.00},\ \mathbf{2.19},\ \mathbf{-57.10})$ | | | | |
| | $\mathbf{z}^{7,4} = (-25700.00,\ 2.32,\ -54.70$ | | | | |
| | $\mathbf{z}^{7,5} = (-30700.00,\ 3.28,\ -45.70)$ | | | | |
| | $\mathbf{z}^{7,6} = (-25600.00,\ 2.26,\ -55.70)$ | | | | |
| | $\mathbf{z}^{7,7} = (-26500.00,\ 2.91,\ -44.60)$ | | | | |
| | $\mathbf{z}^{7,8} = (-26200.00,\ 2.67,\ -48.40)$ | | | | |
| | $\mathbf{z}^{7,9} = (-25800.00,\ 2.39,\ -53.30)$ | | | | |
| | $\mathbf{z}^{7,10} = (-26400.00,\ 2.81,\ -46.20)$ | | | | |

Table 2: Experiment 2: Data provided to the DM and DM's choices in bold face.

# References

[1] K. Miettinen, Nonlinear Multiobjective Optimization, Kluwer Academic Publishers, Boston, 1999.

[2] K. Miettinen, F. Ruiz, A. P. Wierzbicki, Introduction to multiobjective optimization: Interactive approaches, in: J. Branke, K. Deb, K. Miettinen, R. Słowiński (Eds.), Multiobjective Optimization: Interactive and Evolutionary Approaches, Springer, Berlin, Heidelberg, 2008, pp. 27–57.

[3] D. Kahneman, A. Tversky, Prospect theory: An analysis of decision under risk, Econometrica 47 (2) (1979) 263–291.

[4] K. Miettinen, P. Eskelinen, F. Ruiz, M. Luque, NAUTILUS method: An interactive technique in multiobjective optimization based on the nadir point, European Journal of Operational Research 206 (2) (2010) 426–434.

[5] I. Steponavice, S. Ruuska, K. Miettinen, A solution process for simulation-based multiobjective design optimization with an application in the paper industry, Computer-Aided Design 47 (2014) 45–58.

[6] C. A. C. Coello, G. B. Lamont, D. Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems. Second Edition, Springer, New York, 2007.

[7] K. Deb, Multi-objective Optimization using Evolutionary Algorithms, Wiley, Chichester, 2001.

[8] K. Sindhya, K. Miettinen, K. Deb, A hybrid framework for evolutionary multi-objective optimization, IEEE Transactions on Evolutionary Computation 17 (4) (2013) 495–511.

[9] M. Hartikainen, K. Miettinen, M. M. Wiecek, PAINT: Pareto front interpolation for nonlinear multiobjective optimization, Computational Optimization and Applications 52 (3) (2012) 845–867.

[10] K. Klamroth, K. Miettinen, Integrating approximation and interactive decision making in multicriteria optimization, Operations Research 56 (1) (2008) 222–234.

[11] R. C. Purshouse, K. Deb, M. M. Mansor, S. Mostaghim, R. Wang, A review of hybrid evolutionary multiple criteria decision making methods, in: IEEE Congress on Evolutionary Computation (CEC-2014), 2014, pp. 1147–1154.

[12] M. Luque, F. Ruiz, K. Miettinen, Global formulation for interactive multiobjective optimization, OR Spectrum 33 (1) (2011) 27–48.

[13] F. Ruiz, M. Luque, K. Miettinen, Improving the computational efficiency in a global formulation (GLIDE) for interactive multiobjective optimization, Annals of Operations Research 197 (1) (2012) 47–70.

[14] R. Benayoun, J. de Montgolfier, J. Tergny, O. Laritchev, Linear programming with multiple objective functions: Step method (STEM), Mathematical Programming 1 (3) (1971) 366–375.

[15] K. Deb, K. Miettinen, Nadir point estimation using evolutionary approaches: Better accuracy and computational speed through focused search, in: M. Ehrgott, B. Naujoks, T. J. Stewart, J. Wallenius (Eds.), Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems, Springer, Berlin, Heidelberg, 2010, pp. 339–354.

[16] K. Deb., K. Miettinen, S. Chaudhuri, Towards an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches, IEEE Transactions on Evolutionary Computation 14 (6) (2010) 821–841.

[17] P. Korhonen, J. Wallenius, Behavioral issues in MCDM: Neglected research questions, in: J. Climaco (Ed.), Multicriteria Analysis, Springer, Berlin, Heidelberg, 1997, pp. 412–422.

[18] M. Luque, K. Miettinen, P. Eskelinen, F. Ruiz, Incorporating preference information in interactive reference point methods for multiobjective optimization, Omega 37 (2) (2009) 450–462.

[19] A. P. Wierzbicki, The use of reference objectives in multiobjective optimization, in: G. Fandel, T. Gal (Eds.), Multiple Criteria Decision Making, Theory and Applications, Springer, 1980, pp. 468–486.

[20] I. L. Janis, L. Mann, Decision Making: A Psychological Analysis of Conflict, Choice, and Commitment, The Free Press, New York, 1977.

[21] J. N. Morse, Reducing the size of the nondominated set: Pruning by clustering, Computers & Operations Research 7 (1–2) (1980) 55–66.

[22] I. Das, J. E. Dennis, Normal-boundary intersection: A new nethod for generating the Pareto surface in nonlinear multicriteria optimization problems, SIAM Journal on Optimization 8 (3) (1998) 631–657.

[23] A. B. Ruiz, J. M. Cabello, C. A. Platero, F. Blázquez, Multicriteria optimization of the investment in the auxiliary services of thermal power plants: A case study, Energy Conversion and Management, in press.

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

[25] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731.