Incorporation of implicit decision-maker preferences in Multi-Objective Evolutionary Optimization using a multi-criteria classification method

Laura Cruz-Reyes[a], Eduardo Fernandez[b], Patricia Sanchez[c,*], Carlos A. Coello Coello[d], Claudia Gomez[a]

[a] Postgraduate & Research Division, National Mexican Institute of Technology/Madero Institute of Technology, 89440, Tamaulipas, Mexico, lauracruzreyes@itcm.edu.mx, claudia.gomez@itcm.edu.mx

[b] Faculty of Civil Engineering, Autonomous University of Sinaloa, 80040, Sinaloa, Mexico, eddyf@uas.edu.mx

[c] Computer Science in the Graduate Division, National Mexican Institute of Technology/Tijuana Institute of Technology, 22500, Baja California, Mexico. jpatricia.sanchez@gmail.com

[d] Computer Science Department, CINVESTAV-IPN, 07300, Mexico, DF, ccoello@cs.cinvestav.mx

[*] Corresponding author: Blvd. Alberto Limon Padilla y Av. ITR Tijuana S/N, Mesa Otay C.P. 22500, Tijuana, Baja California, Mexico, Tel. (+52) 8333574820 ext. 3111.
E-mail address: jpatricia.sanchez@gmail.com (P. Sanchez)

Abstract

Nowadays, most Multi-Objective Evolutionary Algorithms (MOEA) concentrate mainly on searching for an approximation of the Pareto frontier to solve a multi-objective optimization problem. However, finding this set does not completely solve the problem. The decision-maker (DM) still has to choose the best compromise solution from that set. But as the number of criteria increases, several important difficulties arise in performing this task. Identifying the Region of Interest (ROI), according to the DM's preferences, is a promising alternative that would facilitate the selection process. This paper approaches the incorporation of preferences into a MOEA in order to characterize the ROI by a multi-criteria classification method. This approach is called Hybrid Multi-Criteria Sorting Genetic Algorithm and is composed of two phases. First, a metaheuristic is used to generate a small set of solutions that are classified in ordered categories by the DM. Thus, the DM's preferences will be reflected indirectly in this set. In the second phase, a multi-criteria sorting method is combined with an evolutionary algorithm. The first one is used to classify new solutions. Those classified as 'satisfactory' are used for creating a selective pressure towards the ROI. The effectiveness of our method was proved in nine instances of a public project portfolio problem. The obtained results indicate that our approach achieves a good characterization of the ROI, and outperforms the standard NSGA-II in simple and complex problems. Also, these results confirm that our approach is able to deal with many-objective problems.

Keywords: Evolutionary algorithms; Multi-objective optimization; Implicit preferences; Multi-criteria sorting.

1. Introduction

A wide variety of problems in the real world often involve multiple objectives to be minimized or maximized simultaneously [1]. As a consequence of the conflicting nature of the criteria, it is not possible to obtain a single optimum, and, consequently, the ideal solution to a multi-objective optimization problem (MOP) cannot be reached. As was stated by Fernandez et al. in [2], to solve a MOP means to find the best compromise solution according to the decision maker's (DM) preferences. Multi-Objective Evolutionary Algorithms (MOEAs) have been widely used since the 1990's, standing out in their engineering applications (e.g. [3–11]). MOEAs are particularly attractive to solve MOPs because they deal simultaneously with a set of possible solutions (the MOEA's population), which allows them to obtain an approximation of the Pareto frontier in a single run of the algorithm. However, according to Deb [12] and Fernandez et al. [2], one aspect that is often disregarded in the literature on MOEAs is the fact that the solution of a problem involves not only the search, but also the decision making process. That is, finding the Pareto frontier does not completely solve the problem; the DM still has to choose the best compromise solution out of that set. This is not a difficult task when dealing with problems having two or three objectives. However, as the number of criteria increases, the size of the Pareto frontier increases exponentially. Thus, it becomes harder or even impossible for the DM to establish valid judgments in order to compare many solutions with several conflicting criteria. Besides, the approaches from the field of multi-criteria decision analysis do not perform well on such large decision problems, making it difficult to obtain a single solution [2].

As was stated in Miller's famous paper [13], the capacity of the human mind is restricted to handling a small amount of information at the same time. This is a very serious obstacle when the DM has to compare a subset of non-dominated solutions to identify the best compromise solution in problems with many objectives. Hence, the DM's cognitive effort would be greatly reduced if the MOEA were able to identify the Region of Interest (ROI), the privileged zone of the Pareto frontier that best matches the DM's preferences. The ROI is defined by Deb et al. [14] and Adra et al. [15] as the set of non-dominated solutions that are preferred by the DM over the other solutions. In similar terms, the ROI may be defined as a subset of the Pareto front whose elements are considered satisfactory by the DM. In order to guide the search towards the ROI, the DM would agree with incorporating his/her multi-criteria preferences into the search process. Many approaches have been proposed with this aim; most of them employ an explicit way of incorporating preferences. Here, we propose a method for implicit preference incorporation whose main contribution can be summarized as follows:
Several solutions found by an ant colony metaheuristic are used to capture the preference information from the DM, and with them the DM forms a reference set in which the solutions are assigned to ordered categories. The set of reference examples implicitly contains the DM's belief of what a satisfactory solution is. Thus, the ROI is characterized by this preference information. Then, a multi-criteria classification method is used to assign new solutions to these categories, making selective pressure towards the ROI. This pressure is nearly independent of the number of objective functions, so the quality of solutions is not affected by the increment of the number of objectives. Therefore, the method performs very well in many-objective problems.

The remainder of this paper is structured as follows: Section 2 briefly reviews approaches for incorporating the DM's preferences into multi-objective optimization metaheuristics. In its last part, that section describes a multi-criteria method for assigning solutions to ordered classes (the THESEUS method). With this as a background, our approach of creating a selective pressure towards the ROI is presented in Section 3. Some computer experiments are illustrated in Section 4, and finally, several concluding remarks are discussed in Section 5. The acronyms and notations used along this paper are given in Appendix A.

2. Some background

2.1. A brief outline of preference incorporation in multi-objective evolutionary optimization

Multi-objective metaheuristic approaches have by now demonstrated their ability to approximate the whole Pareto front; however, the number of efficient solutions found is large. The selection of one solution (the final preferred alternative) from among a huge set is evidently a difficult task for the DM, especially when the number of objectives increases [16]. The DM is interested in discovering only the zone of the Pareto front corresponding to his/her preferences (the ROI), instead of the whole Pareto front. It is essential to provide the DM with a small number of satisfactory alternatives, due to the human cognitive limitations referred to by Miller [13]. For this, decision-makers should provide information about their preferences using a representation model. The modeling of the preferences plays a key role in decision-making [17], as it will define the nature and organization of the information. Information about the DM's preferences can be expressed in diverse ways. According to Bechikh [16], the most commonly used approaches are the following:

1. *Weights*. Those in which weighting coefficients are assigned by the DM to each objective function (e.g. [18, 19]).

2. *Ranking solutions*. Those in which the DM performs pair-wise comparisons between pairs of solutions on a subset of the current population, in order to rank the sample's solutions (e.g. [20, 21]).

3. *Ranking objectives*. Those in which the DM performs pair-wise comparisons between pairs of objective functions in order to rank the set of objective functions (e.g. [22, 23]).

4. *Reference point*. Those based on goals or aspiration levels, supplied by the DM, to be achieved by each objective (e.g. [14, 24]).

5. *Reservation point*. Those in which the DM supplies, for each objective, his/her reservation level, that is, the accepted level that (s)he wishes to reach. (e.g. [25]).

6. *Trade-off between objectives*. Those in which the DM specifies acceptable trade-offs between objective functions (e.g. [26]).

7. *Desirability thresholds*. Those in which the DM supplies: *i*) an absolutely satisfying objective value and *ii*) a marginally infeasible objective value, to construct a desirability function (e.g. [27]).

8. *Outranking parameters*. Those in which the DM specifies the necessary weights and thresholds to build a fuzzy outranking relation (e.g. [2, 28]).

A novel approach to an implicit preference incorporation was proposed by Greco et al. in [29]. To the best of our knowledge, they were the first to suggest the use of multi-criteria classification methods combined with an evolutionary algorithm. The preference information is provided by the decision maker in successive iterations by judging some current solutions as 'good' or 'bad'. Using this information, several decision rules are induced through the Dominance-based Rough Set Approach. 'Good' solutions have more fitness than the bad ones. Unfortunately, this interesting idea has not been validated by convincing experiments. In a recent paper, Oliveira et al. [30] used the ELECTRE TRI multi-criteria sorting method combined with an evolutionary algorithm. In ELECTRE TRI a reference profile is introduced to establish the boundary between adjacent ordered categories. ELECTRE TRI has been criticized because defining the reference profiles is often a very hard task, especially when the decision-maker has only a vague idea about the boundary between two consecutive categories. The existence of such boundaries is doubtful in many real-world problems (cf. [31, 32]). Besides, one can question whether a reference profile is sufficient for an acceptable characterization of its related categories. If the object to be sorted were incomparable with several reference profiles, ELECTRE TRI would suggest inappropriate assignments.

The approach by Fernandez et al. [2], the so-called NOSGA2, is an important precedent for our work. The 'a priori' way of incorporating preferences in NOSGA2 has been recently used by Cruz et al. [33] in optimizing interdependent project portfolios with many objectives. In that paper, the authors proposed the Non Outranked Ant Colony Optimization (NO-ACO) method, a short description of which is given below.

2.2. Description of the NO-ACO model

The NO-ACO method uses a set of agents called ants and a local search engine to perform the optimization process. This method incorporates the preferences of the decision maker using the preferences model of Fernandez et al. [2]. The model is based on the relational system of preferences described by Roy in [34]. The essence of the model is the degree of truth of the statement 'x is at least as good as y'. This is represented as $\sigma(x, y)$, and could be calculated using outranking methods such as ELECTRE [35] or PROMETHEE [36]. Let us consider a threshold of acceptable credibility λ, an asymmetry parameter β, and a symmetry parameter ε, where $0 \le \varepsilon \le \beta \le \lambda$ and $\lambda > 0.5$. The model identifies one of the following preference relations for each pair of solutions (x, y):

- *Strict preference*: This corresponds to the situation when the DM has clear and well-defined reasons justifying the choice of x over the other. It is denoted as xPy and defined as a disjunction of the conditions:
  a) x dominates y
  b) $\sigma(x, y) \ge \lambda \wedge \sigma(y, x) < 0.5$ (1)
  c) $\sigma(x, y) \ge \lambda \wedge [0.5 \le \sigma(y, x) < \lambda] \wedge [\sigma(x, y) - \sigma(y, x)] \ge \beta$.

- *Indifference*: From the DM's perspective, this corresponds to the existence of clear and positive reasons that justify an equivalence between the two options. This relationship is denoted as *xIy* and is defined as the conjunction of:
  a) $\sigma(x, y) \geq \lambda \wedge \sigma(y, x) \geq \lambda$ (2)
  b) $|\sigma(x, y) - \sigma(y, x)| \leq \varepsilon$.

- *Weak preference*: This models a state of doubt between *xPy* and *xIy*. It is represented as *xQy* and is defined as the conjunction of:
  a) $\sigma(x, y) \geq \lambda \wedge \sigma(x, y) > \sigma(y, x)$ (3)
  b) $\neg\, xPy \wedge \neg\, xIy$.

- *Incomparability*: From the DM's perspective, there is a high level of heterogeneity between the alternatives, so (s)he cannot express a preference between them. This is denoted as *xRy* and is expressed in terms of:
  $\sigma(x, y)$ as $xRy \Rightarrow \sigma(x, y) < 0.5 \wedge \sigma(y, x) < 0.5$. (4)

- *k-Preference*: This represents a state of doubt between *xPy* and *xRy*. It is represented as *xKy* and is modeled by the conjunction of the following three propositions:
  a) $0.5 \leq \sigma(x, y) < \lambda$
  b) $\sigma(y, x) < 0.5$ (5)
  c) $\sigma(x, y) - \sigma(y, x) > \beta/2$.

Given a set of feasible solutions O, the preferential system of NO-ACO defines the following sets:

- S is composed of the solutions that strictly outrank x, and NS is known as the *non-strictly-outranked frontier*; these sets are defined as follows:
  $S(O, x) = \{y \in O \mid yPx\}$ (6)
  $NS(O) = \{x \in O \mid S(O, x) = \emptyset\}$.

- W is composed of the non-strictly-outranked solutions that weakly outrank x, and NW is known as the *non-weakly-outranked frontier*; these sets are expressed as follows:
  $W(O, x) = \{y \in NS(O) \mid yQx \vee yKx\}$ (7)
  $NW(O) = \{x \in NS(O) \mid W(O, x) = \emptyset\}$.

The net flow score is a popular measure for ranking a set of alternatives on which a fuzzy preference relation is defined (cf. [37]). This measure is used by NO-ACO to identify the DM's preferred solutions on the non-strictly-outranked frontier. It can be defined as:

$$F_n(x) = \sum_{y \in NS(O)\setminus\{x\}} [\sigma(x, y) - \sigma(y, x)]. \tag{8}$$

Since $F_n(x) > F_n(y)$ indicates a preference for x over y, the preferential system defines the following sets:

- F is composed of non-strictly-outranked solutions that are greater in net flow to x, and NF is known as the *net-flow non-outranked frontier*; these sets are defined by:
  $F(O, x) = \{y \in NS(O) \mid F_n(y) > F_n(x)\}$ (9)
  $NF(O) = \{x \in NS(O) \mid F(O, x) = \emptyset\}$.

The problem that NO-ACO solves is

$$\min_{x \in O} \left\{ \left\langle \left|S(O,x)\right|, \left|W(O,x)\right|, \left|F(O,x)\right| \right\rangle \right\} \quad (10)$$

The best solution is found through a lexicographic search, with pre-emptive priority favoring $|S(O, x)|$.

2.3. An outline of the THESEUS method

The aim of the THESEUS method is assigning multi-criteria objects to preference-ordered categories. THESEUS relies on the following premises (cf. [38]):

i. There is a finite set of ordered categories $\boldsymbol{Ct} = \{C_1, \ldots, C_M\}$, $(M \geq 2)$; C_M is assumed to be the preferred category.

ii. Let U be the universe of objects x described by a coherent set of N criteria, denoted $G = \{g_1, g_2, \ldots, g_j, \ldots, g_N\}$, with $N \geq 3$.

iii. There is *a set of reference objects or training examples T*, which is composed of elements $b_{kh} \in U$ assigned to category C_k, $(k = 1,\ldots, M)$.

iv. The decision maker agrees with a fuzzy outranking relation $\sigma(x, y)$ defined on $U \times U$ (see Section 2.2). Its value models the degree of credibility of the statement 'x is at least as good as y' from the decision maker's perspective.

The THESEUS method is based on comparing a new object to be assigned with reference objects through models of preference and indifference relations (cf. [38]). The assignment is not a consequence of the object's intrinsic properties: it is rather the result of comparisons with other objects whose assignments are known. In the following, $C(x)$ denotes a potential category assignment of object x. According to THESEUS, $C(x)$ should satisfy some consistency rules:

$$\forall x \in U, \forall b_{kh} \in T$$

$$xPb_{kh} \Rightarrow C(x) \succsim C_k \quad (11.a)$$
$$b_{kh}Px \Rightarrow C_k \succsim C(x)$$

$$xQb_{kh} \Rightarrow C(x) \succsim C_k \quad (11.b)$$
$$b_{kh}Qx \Rightarrow C_k \succsim C(x)$$

$$xIb_{kh} \Rightarrow (C(x) \succsim C_k) \wedge (C_k \succsim C(x)) \Leftrightarrow C(x) = C_k \quad (11.c)$$

Relations P, Q, I were defined in Eqs. (1–3). The symbol $\succsim$ denotes the statement 'is not worse than' on the set of categories, which is related to the decision-aiding context. Note that $C(x)$ is a variable whose domain is the set of ordered categories. Eqs. (11.a–c) express the necessary consistency amongst the preference model, the reference set and the appropriate assignments of x. The assignment $C(x)$ should be as compatible as possible with the current knowledge about the assignment policy.

THESEUS uses the inconsistencies with Eqs. (11.a–c) to compare the possible assignments of x; more specifically:

- The set of P-inconsistencies for x and $C(x)$ is defined as $D_P = \{(x,b_{kh}), (b_{kh},x), b_{kh} \in T$ such that (11.a) is FALSE$\}$;
- The set of Q-inconsistencies for x and $C(x)$ is defined as $D_Q = \{(x,b_{kh}), (b_{kh},x), b_{kh} \in T$ such that (11.b) is FALSE$\}$;
- The set of I-inconsistencies for x and $C(x)$ is defined as $D_I = \{(x,b_{kh}), (b_{kh},x), b_{kh} \in T$ such that (11.c) is FALSE$\}$.

Suppose that $C(x) = C_k$ and consider $b_{jh} \in T$. Some cases in which $xIb_{jh} \wedge |k-j| = 1$ might be explained by 'discontinuity' of the description are: x may be close to the upper (lower) boundary of C_k and b_{jh} may be close to the lower (upper) boundary of C_j. They will be called second-order I-inconsistencies and grouped in the set D_{2I}. The set $D_{1I} = D_I - D_{2I}$ contains the so-called first-order I-inconsistencies, which are not consequences of the described discontinuity effect. Let n_P, n_Q, n_{1I}, n_{2I} denote the cardinalities of the above-defined inconsistency sets, $N_1 = n_P + n_Q + n_{1I}$, and $N_2 = n_{2I}$.
THESEUS suggests an assignment that minimizes the above inconsistencies with lexicographic priority favouring N_1, which is the most important criterion [38]. The basic assignment rule is

For each $x \in U$ and given a minimum credibility level $\lambda > 0.5$,
a) starting with $k = 1$ ($k = 1,\ldots M$) and considering each $b_{kh} \in T$, calculate $N_1(C_k)$;
b) identify the set $\{C_j\}$ whose elements hold $C_j = argmin\, N_1(C_k)$;
c) select $C_{k*} = \underset{\{C_j\}}{argmin}\, N_2(C_i)$.
d) If C_{k*} is a single-category solution, assign x to C_{k*}; other situations are approached as below.

The suggestion may be a single category or a sequence of categories. The first case is called a well-defined assignment; otherwise, the obtained multi-category solution highlights the highest category (C_H) and the lowest category (C_L); the latter is appropriate for assigning the object, but fails in determining the most appropriate. Such solution will be called 'a vague assignment'.

3. An approach to finding satisfactory solutions using a multi-criteria sorting method

In this paper, we present a new idea to incorporate preferences in an implicit way into multi-objective evolutionary optimization. Our approach consists in using a set of solutions that have been ordered by the DM on a set of categories. These assignments represent knowledge about the DM's preferences. Applying this knowledge in the framework of a multi-criteria sorting method, every new solution created by the search process can be assigned to one of the already stated categories. To some extent, such preference information replaces the DM when classifying new solutions. Here, we present a way of using this capacity to create a selective pressure towards the ROI. We propose a method called Hybrid Multi-Criteria Sorting Genetic Algorithm (H-MCSGA). This approach is composed of two phases, which are described in the following subsections. The complete procedure of the H-MCSGA is shown below:

PROCEDURE H-MCSGA (*L*, Number_of_Generations)
Run a multi-objective optimization method and obtain an approximation to the Pareto frontier *PF*
Initialize the reference set *T* by using a subset of *PF* (which satisfies the self-consistent property)
Set σ-parameters agreeing with *T*
Initialize Parent Population *P* including the elements of *T*
Complete *P* with random individuals so that it reaches a size of *L*.
Generate non-dominated fronts on *P* (based on evaluation of the values of the objective function)
Assign to these fronts a rank (level) F_i
Calculate σ on $F_1 \times T$
For each $x \in F_1$, assign *x* to one preferred category C_k using σ
Form *M'* sub-fronts of F_1
Assign to each sub-front of *M'* a rank (level) and update the levels of the remaining fronts
Generate from *P* a Child Population *Q* of size *L*
    Perform Binary Tournament Selection (based on non-domination rank)
    Perform Recombination and mutation
FOR I = 1 to Number_of_Generations DO
    Assign $P' = P \cup Q$
    Generate non-dominated fronts on *P'*
    Assign to these fronts a rank (level) F_i
    Calculate σ on $F_1 \times T$
    For each $x \in F_1$, assign *x* to one preferred category C_k using σ
    Form *M'* sub-fronts of F_1
    Assign to each sub-front of *M'* a rank (level) and update the levels of the remaining fronts
    Repeat to each front until size of new population $|P| \geq L$
        Calculate crowding distance for each solution in the current front
        Add these solutions to new population *P*
    Sort *P* using non-domination rank and crowding distance
    Keep in *P* the first *L* individuals
    Generate from *P* a Child Population *Q* of size *L*
        Perform Binary Tournament Selection (based on non-domination rank and crowding distance)
        Perform Recombination and mutation
End FOR
End PROCEDURE

3.1. A method to construct a reference set (Phase 1)

In the first phase of the H-MCSGA, a multi-objective metaheuristic approach will be used to obtain an approximation to the Pareto frontier; that set of solutions will be sorted by the DM, into a set of categories to construct the reference set. Thus, the DM's preferences will be reflected in it. We chose the NO-ACO algorithm (Section 2.2) because, unlike Pareto-based evolutionary algorithms, it has proved its ability to find good solutions to problems with high dimensions [33]. To construct the reference set, the solutions obtained by NO-ACO were categorized by simulating a DM whose preferences are compatible with the outranking model from [2]. The categories considered to form the reference set are 'Satisfactory' and 'Unsatisfactory'. The reference set is constructed according to the following steps:

1. Run NO-ACO to find a set *A* of solutions containing a subset of the approximate Pareto frontier.
2. Create the 'Satisfactory' category with those solutions belonging to the known Pareto frontier that fulfil at least one of the following conditions: *a*) to be the solution of highest net flow (Eq. (8)) on the known Pareto frontier; or *b*) to be a solution that satisfies $|S(A, x)| = 0$ (Eq. (6)) and $|W(A, x)| = 0$ (Eq. (7)), that is, that belongs to the non-strictly and non-weakly outranked frontier in *A*.
3. Create the 'Unsatisfactory' category with the remaining solutions generated by NO-ACO in step 1.

In cases where the satisfactory category of the reference set is poorly populated, it is necessary to add fictitious solutions (derived from an existing solution) to extend and intensify the category. Each fictitious solution has a high degree of indifference to the solution from which it is derived. The procedure for generating them is as follows: *a*) to identify a pair of objectives with nearly equal weights, called *similar weight objectives*; *b*) to create a replica of an existing solution; and *c*) to modify the *similar weight objectives* of the replicated solution, adding to one of them and subtracting to the other one the same predefined value. We do this in order to make a slight variation in the objectives of the existing solution, in the sense of improvement and compensation. An example is given in Table 1.

Table 1
Real and fictitious solutions.

Reference Element	Objective values				Category
	N_1	N_2	N_3	N_4	
3	**1306630**	1023530	**1598110**	994340	Satisfactory (real)
4	**1307130**	1023530	**1597610**	994340	Satisfactory (fictitious)

3.2. A method to search the ROI (Phase 2)

The second phase of the H-MCSGA is a variant of the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [39], but incorporating the THESEUS method (Section 2.3). We chose THESEUS because it has given good results with artificial and real-world data (cf. [38]). In comparison with other multi-criteria sorting methods, THESEUS can handle more general and larger reference sets, thus providing more suitable assignments (cf. [40]). Because of its generality and other good properties, THESEUS is used in the present paper as a classification tool in order to characterize the ROI. Our method uses the reference set constructed in the first phase, for finding satisfactory solutions in the search process. This approach works like the NSGA-II but with the following additional steps (see Fig. 1):

a. Each solution of the non-dominated front of NSGA-II (the first front) is assigned by THESEUS to one category of the set {Satisfactory, Perhaps Satisfactory, Unsatisfactory}, where 'Perhaps Satisfactory' corresponds to a vague assignment performed by THESEUS;
b. The first front of NSGA-II is divided in three sub-fronts; the first ranked sub-front contains the solutions that were assigned to the most preferred category ('Satisfactory');

c. The remaining fronts of the current NSGA-II population are re-ordered by considering each sub-front of the original non-dominated front as a new front;
d. The same actions of NSGA-II are implemented, but considering the new fronts; particularly, the NSGA-II's elitism involves the new first front, which is now constituted by the non-dominated solutions belonging to the most preferred category.

In an MOP, the ROI should be composed of solutions that belong to the most preferred category. Hence, the solutions in the ROI are characterized by the fact that they are: *i*) non-dominated and *ii*) considered satisfactory ones by the DM. Therefore, our approach creates a selective pressure towards solutions that have both features.

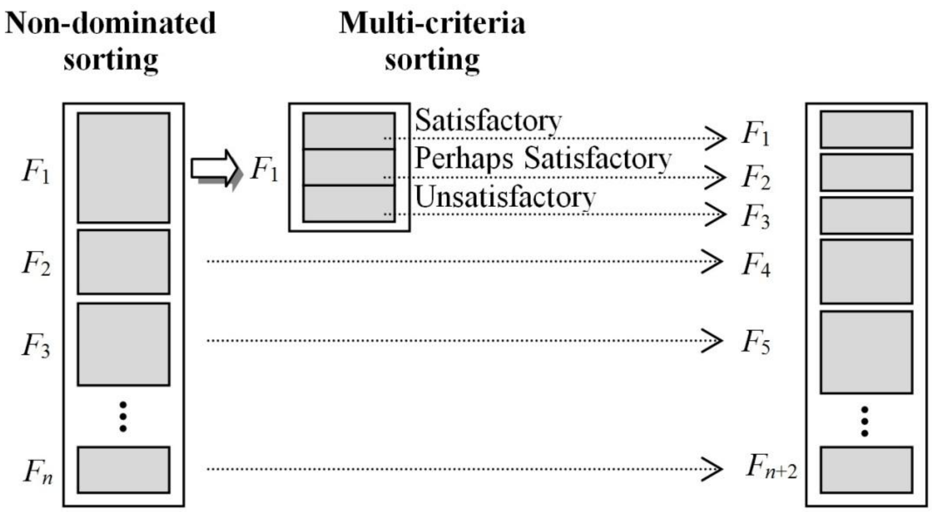

Fig. 1. Multi-criteria sorting by H-MCSGA.

4. Some computer experiments

4.1. Case study: A Public Project Portfolio Problem (PPPP)

Let us consider a decision making situation in which the DM is in charge of selecting a group of projects (portfolio) that her/his institution will implement. The aim of this decision problem is to choose the 'best' portfolio satisfying some budget constraints. Formalizing these concepts, let us consider a set of N projects, where the ith project is represented by a p-dimensional vector $f(i) = \langle f_1(i), f_2(i), f_3(i), \ldots, f_p(i)\rangle$, where each $f_j(i)$ indicates the contribution of project i to the jth objective. Each objective denotes the benefit target; that is, people belonging to a social category (e.g. Extreme Poverty, Poverty, Middle), who receive a benefit level (e.g. High Impact, Middle Impact, Low Impact) from the ith project.

On the other hand, a portfolio x is a subset of these projects which is usually modeled as a binary vector $x = \langle x_1, x_2, \ldots, x_N\rangle$. In this vector, x_i is a binary variable where $x_i = 1$ if the ith project is supported and $x_i = 0$ otherwise.

There is a total budget that the organization is willing to invest, which is denoted as B; each project has an associated cost c_i. Portfolios are subject to the budget constraint:

$$\left(\sum_{i=1}^{N} x_i c_i\right) \leq B \tag{12}$$

The ith project corresponds to an area (e.g. health, education) denoted by a_i. Each area has budgetary limits defined by the DM or any other competent authority. Let us consider for each area k, a lower and an upper limit, L_k and U_k respectively. Based on this, the constraint for each area k is

$$L_k \leq \sum_{i=1}^{N} x_i g_i(k) c_i \leq U_k \tag{13}$$

where g is defined as

$$g_i(k) = \begin{cases} 1 & if \quad a_i = k, \\ 0 & otherwise \end{cases} \tag{14}$$

Besides, each project corresponds to a geographical region which it will benefit. In the same way as areas, each region has lower and upper limits as another constraint that must be fulfilled by a feasible portfolio.

The quality of a portfolio x is determined by the union of the benefits of each of the projects that compose it. This can be expressed as

$$z(x) = \langle z_1(x), z_2(x), z_3(x), ..., z_p(x) \rangle \tag{15}$$

where $z_j(x)$ in its simplest form, is defined as

$$z_j(x) = \sum_{i=1}^{N} x_i f_j(i) \tag{16}$$

If we denote by R_F to the region of feasible portfolios, the problem of the project portfolio is to identify one or more portfolios that solve

$$\max_{x \in R_F} \{z(x)\} \tag{17}$$

In this problem, the only accepted solutions are those portfolios that satisfy the following constraints: the total budget constraint (Eq. (12)), area constraints (Eq. (13)), and region constraints (similar to Eq. (13)).

4.2. Experimental design

Let us consider two experiments addressing the Problem (17) with H-MCSGA. In both experiments, we want to verify that our approach makes a good characterization of the ROI. In addition, we conduct the first experiment to determine if the H-MCSGA is capable of improving the solutions obtained by NO-ACO during the first phase; the second experiment explores whether our method outperforms NSGA-II. The NO-ACO parameters in the first phase of the H-MCSGA were the same as those reported in [33], but we neither used local search nor considered synergy. The parameters of the evolutionary search in the second phase of the H-MCSGA were: crossover probability = 1; mutation probability = 0.05; population size = 100, number of generations = 500. The parameters of the preference model and the data of the projects are different for each instance. The credibility threshold was set as $\lambda = 0.67$. H-MCSGA was programmed in the Java language, using the JDK 1.7 compiler, and NetBeans 7.1 as IDE. The algorithms were run 30 times for each instance on an Intel CORE i5, 2.5 GHz processor with 4 GB of RAM.

4.3. First experiment: NO-ACO vs H-MCSGA

The first experiment was designed to show that H-MCSGA can improve the solutions obtained by NO-ACO, which are part of the reference set used by the second phase of our method. The experiment consisted in comparing the quality of the solutions provided by H-MCSGA after its two phases against those obtained from NO-ACO during the first phase of the H-MCSGA. We experimented with six random instances whose basic information is shown in Table 2; in each run a different reference set was used. The performance of our approach is illustrated at the instance level and extensive results are shown for only three instances in Tables 3–5.

Table 2
Information about instances used in the first experiment.

Instance	Instance Description	
	Objectives	Projects
1	9	100
2	9	100
3	9	100
4	9	150
5	9	150
6	16	500

Analyzing the results of Table 3, we can see that the size of solution sets from H-MCSGA provides a better representation of the ‘Satisfactory’ category in comparison with NO-ACO (Columns 2 and 3). That is, H-MCSGA provides a rich characterization of the ROI. On the other hand, H-MCSGA always maintains its solutions as non-dominated (Columns 3 and 5). Furthermore, in 28/30 times, the second phase of the H-MCSGA found a solution which improves, in net flow score, the solution previously obtained by its first phase (running NO-ACO) (Columns 6 and 7). Consequently, the solutions generated by H-MCSGA outperform the NO-ACO solutions.

Table 3
Comparative results between NO-ACO and H-MCSGA (instance 1 in Table 2).

Run	Size of the solution set (satisfactory solutions)		Non-dominated solutions in $(A \cup B)$[a]		The highest net flow is in	
	A	*B*	From *A*	From *B*	*A*	*B*
1	3	105	2	105	×	✓
2	3	90	3	90	×	✓
3	3	42	3	42	×	✓
4	3	79	3	79	×	✓
5	3	70	3	70	×	✓
6	3	92	3	92	×	✓
7	3	86	3	86	×	✓
8	4	94	4	94	✓	✓

9	3	74	3	74	×	✓
10	3	76	3	76	×	✓
11	3	89	3	89	×	✓
12	3	92	3	92	×	✓
13	3	90	3	90	×	✓
14	3	95	3	95	×	✓
15	2	18	2	18	×	✓
16	3	23	3	23	×	✓
17	3	86	3	86	×	✓
18	3	92	3	92	×	✓
19	4	72	4	72	×	✓
20	3	68	3	68	×	✓
21	3	71	3	71	×	✓
22	3	97	3	97	×	✓
23	3	71	3	71	×	✓
24	3	78	3	78	×	✓
25	3	94	3	94	×	✓
26	3	91	3	91	×	✓
27	3	56	3	56	×	✓
28	3	92	3	92	✓	✓
29	4	91	4	91	×	✓
30	5	97	5	97	×	✓

[a] *A* is the set of solutions obtained by NO-ACO during the first phase of the H-MCSGA; *B* is the set obtained by the second phase of the H-MCSGA.

In Table 4 we can see again that the hybrid procedure provides a rich characterization of the ROI as compared to NO-ACO (Columns 2 and 3). On the other hand, both algorithms have the same performance in terms of Pareto dominance because they hold their solutions as non-dominated (Columns 2 and 4 for NO-ACO; 3 and 5 for H-MCSGA). However, the second phase of the H-MCSGA always improves NO-ACO in net flow score (Columns 6 and 7).

Table 4
Comparative results between NO-ACO and H-MCSGA (instance 3 in Table 2).

Run	Size of the solution set (satisfactory solutions)		Non-dominated solutions in ($A \cup B$)[a]		The highest net flow is in	
	A	*B*	From *A*	From *B*	*A*	*B*
1	3	65	3	65	×	✓
2	3	79	3	79	×	✓
3	3	81	3	81	×	✓
4	3	44	3	44	×	✓
5	3	68	3	68	×	✓
6	3	73	3	73	×	✓
7	3	72	3	72	×	✓
8	3	92	3	92	×	✓
9	3	82	3	82	×	✓
10	3	89	3	89	×	✓
11	3	55	3	55	×	✓
12	3	55	3	55	×	✓
13	3	94	3	94	×	✓
14	3	52	3	52	×	✓
15	3	62	3	62	×	✓
16	1	33	1	33	×	✓
17	3	72	3	72	×	✓
18	3	78	3	78	×	✓
19	3	71	3	71	×	✓
20	4	90	4	90	×	✓

21	3	82	3	82	×	✓
22	3	94	3	94	×	✓
23	3	80	3	80	×	✓
24	3	53	3	53	×	✓
25	3	78	3	78	×	✓
26	4	91	4	91	×	✓
27	3	84	3	84	×	✓
28	3	73	3	73	×	✓
29	3	94	3	94	×	✓
30	3	60	3	60	×	✓

[a] *A* is the set of solutions obtained by NO-ACO during the first phase of the H-MCSGA; *B* is the set obtained by the second phase of the H-MCSGA.

The information provided in Table 5 shows that H-MCSGA in 27/30 times provides a better representation of the ROI in comparison with NO-ACO (Columns 2 and 3). On the other hand, H-MCSGA always maintains its solutions as non-dominated whilst NO-ACO does it only 19/30 times (Columns 2 and 4 for NO-ACO; 3 and 5 for H-MCSGA). Moreover, in 21/30 times, the second phase of the H-MCSGA improves NO-ACO in net flow score (Columns 6 and 7).

Table 5
Comparative results between NO-ACO and H-MCSGA (instance 6 in Table 2).

Run	Size of the solution set (satisfactory solutions)		Non-dominated solutions in ($A \cup B$)[a]		The highest net flow is in	
	A	*B*	From *A*	From *B*	*A*	*B*
1	11	94	11	94	×	✓
2	17	44	17	44	×	✓
3	10	90	10	90	×	✓
4	9	9	9	9	✓	✓
5	7	97	5	97	×	✓
6	9	101	8	101	×	✓
7	9	92	8	92	×	✓
8	9	73	9	73	×	✓
9	16	90	15	90	✓	✓
10	8	92	8	92	✓	✓
11	14	91	12	91	×	✓
12	8	90	8	90	×	✓
13	11	90	10	90	×	✓
14	8	63	7	63	×	✓
15	13	88	12	88	×	✓
16	15	89	14	89	✓	✓
17	16	85	16	85	×	✓
18	14	87	14	87	×	✓
19	12	36	11	36	✓	✓
20	9	6	9	6	✓	✓
21	11	70	11	70	×	✓
22	8	54	8	54	×	✓
23	11	91	10	91	×	✓
24	7	19	7	19	×	✓
25	12	43	12	43	×	✓
26	15	86	15	86	×	✓
27	12	43	12	43	✓	✓
28	12	88	12	88	×	✓
29	14	12	14	12	✓	✓
30	10	91	10	91	✓	✓

[a] *A* is the set of solutions obtained by NO-ACO during the first phase of the H-MCSGA; *B* is the set obtained by the second phase of the H-MCSGA.

The detailed results of the other instances are not significantly different from those previously shown. This can be observed in Table 6, which summarizes the results of the 30 runs carried out for each instance. Based on these results, we may conclude that H-MCSGA gives on average for each instance a better characterization of the ROI than NO-ACO (Columns 2 and 3). Moreover, the H-MCSGA always holds their solutions as non-dominated whilst sometimes the NO-ACO solutions are dominated (Columns 4 and 5). On the other hand, in at least 21/30 times, the second phase of the H-MCSGA found a solution that improves, in net flow score, the solution previously obtained by its first phase (Columns 6 and 7).
The results support that the H-MCSGA is able to: *a*) make a good characterization of the ROI, and *b*) in most of the times, improve the solutions previously obtained by its first phase.

Table 6
Summary of comparative results between NO-ACO and H-MCSGA.

Instance	Average size of the solution set (satisfactory solutions)		Times that the solution set remains non-dominated in ($A \cup B$)[a]		Times that the highest net flow is in	
	A	*B*	From *A*	From *B*	*A*	*B*
1	3	79	29	30	2	30
2	3	59	29	30	7	30
3	3	73	30	30	0	30
4	4	69	27	30	8	30
5	2	38	26	30	8	30
6	11	70	19	30	9	30

[a] *A* is the set of solutions obtained by NO-ACO during the first phase of the H-MCSGA; *B* is the set obtained by the second phase of the H-MCSGA.

4.4. Second experiment: NSGA-II vs H-MCSGA

The second experiment was designed to show that H-MCSGA outperforms NSGA-II, even in relatively simple problems in which the second is expected to perform well. We experimented with eight random instances described in Table 7; for each instance, one reference set was created. The results are summarized in Table 8.

Table 7
Information about instances used in the second experiment.

Instance	Instance Description	
	Objectives	Projects
1	4	25
2	4	25
3	4	25
4	9	100
5	9	100
6	9	100
7	9	150
8	9	150

Table 8
Comparative results between NSGA-II and H-MCSGA.

Instance	Algorithm	Average		
		Size of the solution set	Solutions that remain non-dominated in $(A \cup B)$[a]	Non-dominated solutions belonging to the satisfactory category (the approximated ROI)
1	NSGA-II	95	85	1
	H-MCSGA	26	26	26
2	NSGA-II	84	79	7
	H-MCSGA	20	20	20
3	NSGA-II	91	90	2
	H-MCSGA	3	3	3
4	NSGA-II	118	48	0
	H-MCSGA	96	96	96
5	NSGA-II	122	69	0
	H-MCSGA	57	57	57
6	NSGA-II	120	61	0
	H-MCSGA	60	60	60
7	NSGA-II	113	22	0
	H-MCSGA	96	96	96
8	NSGA-II	110	27	0
	H-MCSGA	100	100	100

[a] A is the set of solutions obtained by NSGA-II and B is the set obtained by H-MCSGA.

Instances 1–3 correspond to relatively simple problems. The results reveal that on average our approach dominates between 1%–11% of the solutions suggested by NSGA-II, while the solutions from H-MCSGA always remain non-dominated (Column 4). Only a few non-dominated solutions from NSGA-II belong to the 'Satisfactory' category, whereas all H-MCSGA non-dominated solutions are satisfactory (Column 5). Besides, in most of the instances, our method obtained a richer characterization of the ROI (Column 5).

Instances 4–8 are more complex with many objectives and projects. It is not surprising to note a degraded NSGA-II performance, more critical in Instances 7–8. We can see that the solutions from our approach dominate 43%–81% of the solutions suggested by NSGA-II, whereas no H-MCSGA solution is dominated by any solution generated by NSGA-II (Column 4). Also, there are no NSGA-II non-dominated solutions belonging to the 'Satisfactory' category, while our approach always finds many satisfactory solutions (Column 5). Every solution from H-MCSGA belongs to the known ROI, according to the revealed DM preferences.
Based on these results, we can conclude that the H-MCSGA is able to: *a*) make a good characterization of the ROI, and *b*) outperform the NSGA-II even in relatively simple instances.

4.5. Identifying the best compromise

The solutions in the ROI satisfy a necessary condition to be the best compromise, but only one can be chosen. In both experiments performed, the set of satisfactory solutions generated by the H-MCSGA may be still too large to consider that the selection process is a simple task for the DM. Any satisfactory solution might be chosen, but probably there are some that are a little more desirable. In order to help the DM to obtain the best compromise, we need a method to select a subset of solutions containing the 'best' alternatives: this is the choice problematic (P.α) defined by Bernard Roy in [34]. As a contribution to this problem, we could show the DM a subset of satisfactory solutions belonging to the non-strictly-outranked frontier (Eq. (6)). Furthermore, we could show the net flow (Eq. (8)) of these solutions, as extra information to make the selection process even easier.

5. Conclusions and future work

We have presented an original idea for incorporating implicit preferences in multi-objective evolutionary optimization. Our approach, called Hybrid Multi-Criteria Sorting Genetic Algorithm, creates a selection pressure towards the Region of Interest (ROI) instead of the whole Pareto front. Here, solutions belonging to the ROI are characterized by two properties: *i*) Pareto optimality and *ii*) being considered by the DM as satisfactory solutions.

The DM's preferences are captured in a training set formed by solutions assigned to preference-ordered categories. These solutions belong to the known Pareto frontier that is generated by a multi-objective metaheuristic algorithm during the first phase of our approach. Although any metaheuristic may be used in the first phase, we used the NO-ACO algorithm. The second phase of our method is an adaptation of the NSGA-II; here the DM's preferences are articulated when new generated solutions are assigned to ordered categories by the THESEUS multi-criteria sorting method. Once this second phase is finished, an 'a posteriori' articulation of preferences is also needed in order to choose the final best compromise in the ROI.

In examples with 9 and 16 objectives, our method improves solutions obtained by NO-ACO, achieving a better characterization of the preferred category, in terms of number and quality of solutions. In examples with 4 and 9 objectives, our approach finds solutions that outperform NSGA-II, both in terms of Pareto dominance and in terms of characterizing the ROI.

This work may have some advantages, mainly in problems with many objectives: first, the selection pressure towards the ROI is increased, thus achieving a better characterization of this privileged zone of the Pareto frontier. Second, unlike Pareto-based evolutionary algorithms, the number of 'good' solutions in a current population does not significantly depend on the dimension of the objective space; hence, the selection pressure does not decrease with the number of objective functions. Third, a reduction of the cognitive effort from the DM when (s)he must finally choose the best solution among a limited set of satisfactory solutions. Other advantage of our work is the use of methodologies that have dealt with high-dimensional problems (NO-ACO) and that have proved its good performance with artificial and real-world data (THESEUS). Besides, unlike our work, the preceding approaches in [29, 30] do not address the first phase in which the reference set is created. They assume that, from the beginning, the DM can provide his/her judgments about solutions (in fact, his/her 'concept' of satisfactory solution) without knowing any reference result; this is a very difficult task. In our approach, only after the first phase, the DM defines what a satisfactory solution is.

As immediate work, we are going to develop an interactive (progressive) way of incorporating the DM's preferences; his/her preferences may be frequently updated as the second phase progresses. The DM could update his/her idea of what a satisfactory solution is. This could have the advantage of the 'learning' process that is typical of the interactive multi-objective methods. Moreover, we are planning to extend the applicability of our approach to a wider multi-objective optimization context.

Acknowledgements

We acknowledge the support from PROMEP through the network project 'Optimization and Decision Support'. We also thank to DGEST and the Autonomous University of Sinaloa (PROFAPI-055/2012 project).

References

[1] C.A. Coello Coello, G.B. Lamont, D.A. Van Veldhuizen, Evolutionary algorithms for solving multi-objective problems, second ed., Springer, New York, 2007.

[2] E. Fernandez, E. Lopez, F. Lopez, C.A. Coello Coello, Increasing selective pressure towards the best compromise in evolutionary multiobjective optimization: The extended NOSGA method, Information Sciences 181 (2011) 44–56. doi:10.1016/j.ins.2010.09.007.

[3] A. Ghaffarizadeh, K. Ahmadi, M. Eftekhari, Adding Crossover to Extinction-Based Evolutionary Algorithms, Second International Conference on Computer and Electrical Engineering, ICCEE '09, 2 (2009) 43–48. doi:10.1109/ICCEE.2009.125.

[4] A. Kornelakis, Multiobjective Particle Swarm Optimization for the optimal design of photovoltaic grid-connected systems, Solar Energy, 84 (2010) 2022–2033. doi:10.1016/j.solener.2010.10.001.

[5] E.G. Carrano, C.G. Tarôco, O.M. Neto, R.H.C. Takahashi, A multiobjective hybrid evolutionary algorithm for robust design of distribution networks, International Journal of Electrical Power & Energy Systems 63 (2014) 645–656. doi:10.1016/j.ijepes.2014.06.032.

[6] A. Ghaffarizadeh, M. Eftekhari, A.K. Esmailizadeh, N.S Flann, Quantitative Trait Loci Mapping Problem: An Extinction-Based Multi-Objective Evolutionary Algorithm Approach, Algorithms 6(3) (2013) 546–564. doi:10.3390/a6030546.

[7] X. Pelet, D. Favrat, G. Leyland, Multiobjective optimisation of integrated energy systems for remote communities considering economics and CO2 emissions, International Journal of Thermal Sciences 44(12) (2005) 1180–1189. doi:10.1016/j.ijthermalsci.2005.09.006.

[8] H. Bilil, G. Aniba, M. Maaroufi, Multiobjective Optimization of Renewable Energy Penetration Rate in Power Systems, Energy Procedia 50 (2014) 368–375. doi:10.1016/j.egypro.2014.06.044.

[9] P. Ahmadi , I. Dincer, M.A. Rosen, Thermoeconomic multi-objective optimization of a novel biomass-based integrated energy system, Energy 68 (2014) 958–970. doi:10.1016/j.energy.2014.01.085.

[10] P. Ahmadi, I. Dincer, M.A. Rosen, Exergy, exergoeconomic and environmental analyses and evolutionary algorithm based multi-objective optimization of combined cycle power plants, Energy 36(10) (2011) 5886–5898. doi:10.1016/j.energy.2011.08.034.

[11] P. Ahmadi , I. Dincer, M.A. Rosen, Thermodynamic modeling and multi-objective evolutionary-based optimization of a new multigeneration energy system, Energy Conversion and Management 76 (2013) 282–300. doi:10.1016/j.enconman.2013.07.049.

[12] K. Deb, Current trends in evolutionary multi-objective optimization, International Journal for Simulation and Multidisciplinary Design Optimization 1 (2007) 1–8. doi:10.1051/ijsmdo:2007001.

[13] G.A. Miller, The magical number seven, plus or minus two: some limits on our capacity for processing information, Psychological review 63 (1956) 81–97.

[14] K. Deb, J. Sundar, N. Udaya Bhaskara Rao, S. Chaudhuri, Reference point based multi-objective optimization using evolutionary algorithms, International Journal of Computational Intelligence Research 2 (2006) 273–286.

[15] S.F. Adra, I. Griffin, P.J. Fleming, A comparative study of progressive preference articulation techniques for multiobjective optimisation, Evolutionary Multi-Criterion Optimization, Springer, Japan, 2007, pp. 908–921. doi:10.1007/978-3-540-70928-2_67.

[16] S. Bechikh, Incorporating decision maker's preference information in evolutionary multi-objective optimization, PhD thesis, University of Tunis, Tunisia, 2013.

[17] P.J. Sanchez, Modelos para la combinación de preferencias en toma de decisiones: herramientas y aplicaciones, PhD thesis, University of Granada, Spain, 2007.

[18] J. Branke, K. Deb, Integrating user preferences into evolutionary multi-objective optimization, Knowledge Incorporation in Evolutionary Computation, Springer, 2005, pp. 461–477. doi:10.1007/978-3-540-44511-1_21.

[19] E. Zitzler, D. Brockhoff, L. Thiele, The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration, in: Evolutionary Multi-Criterion Optimization, Springer, Matsushima, Japan, 2007, pp. 862–876. doi:10.1007/978-3-540-70928-2_64.

[20] J.W. Fowler, E.S. Gel, M.M. Köksalan, P. Korhonen, J.L. Marquis, J. Wallenius, Interactive evolutionary multi-objective optimization for quasi-concave preference functions, European Journal of Operational Research 206 (2010) 417–425. doi:10.1016/j.ejor.2010.02.027.

[21] K. Deb, A. Sinha, P.J. Korhonen, J. Wallenius, An interactive evolutionary multiobjective optimization method based on progressively approximated value functions,

IEEE Transactions on Evolutionary Computation 14 (2010) 723–739. doi:10.1109/TEVC.2010.2064323.

[22] D. Cvetkovic, I.C. Parmee, Preferences and their application in evolutionary multiobjective optimization, IEEE Transactions on Evolutionary Computation 6 (2002) 42–57. doi:10.1109/4235.985691.

[23] L. Rachmawati, D. Srinivasan, Incorporating the notion of relative importance of objectives in evolutionary multiobjective optimization, IEEE Transactions on Evolutionary Computation 14 (2010) 530–546. doi:10.1109/TEVC.2009.2036162.

[24] S. Bechikh, S.L. Ben, K. Ghédira, The r-dominance: a new dominance relation for preference-based evolutionary multi-objective optimization, Technical Report BS-2010-001, SOIE Research Unit, High Institute of Management of Tunis, Tunisia, 2010.

[25] K. Deb, A. Kumar, Interactive evolutionary multi-objective optimization and decision-making using reference direction method, in: Proceedings of the 9th annual conference on genetic and evolutionary computation, ACM, New York, NY, USA, 2007, pp. 781–788. doi:10.1145/1276958.1277116.

[26] J. Branke, T. Kaußler, H. Schmeck, Guidance in evolutionary multi-objective optimization, Advances in Engineering Software 32 (2001) 499–507. doi:10.1016/S0965-9978(00)00110-1.

[27] T. Wagner, H. Trautmann, Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions, IEEE Transactions on Evolutionary Computation 14 (2010) 688–701. doi:10.1109/TEVC.2010.2058119.

[28] E. Fernandez, E. Lopez, S. Bernal, C.A. Coello Coello, J. Navarro, Evolutionary multiobjective optimization using an outranking-based dominance generalization, Computers & Operations Research 37 (2010) 390–395. doi:10.1016/j.cor.2009.06.004.

[29] S. Greco, B. Matarazzo, R. Słowiński, Interactive evolutionary multiobjective optimization using dominance-based rough set approach, IEEE Congress on Evolutionary Computation (2010) 1–8. doi:10.1109/CEC.2010.5585982.

[30] E. Oliveira, C.H. Antunes, A. Gomes, A comparative study of different approaches using an outranking relation in a multi-objective evolutionary algorithm, Computers & Operations Research 40(6) (2013) 1602–1615. doi:10.1016/j.cor.2011.09.023.

[31] M. Köksalan, V. Mousseau, Ö. Özpeynirci, , S.B. Özpeynirci, A new outranking-based approach for assigning alternatives to ordered classes, Naval Research Logistics 56(1) (2009) 74–85. doi:10.1002/nav.20315.

[32] J. Almeida-Dias, J. Figueira, B. Roy, Electre Tri-C: a multiple criteria sorting method based on characteristic reference actions, European Journal of Operational Research 204 (2010) 565–580. doi:10.1016/j.ejor.2009.10.018.

[33] L. Cruz, E. Fernandez, C. Gomez, G. Rivera, F. Perez, Many-objective portfolio optimization of interdependent projects with ‘a priori’ incorporation of decision-maker preferences, Appl. Math. Inf. 8 (2014) 1517–1531. doi:10.12785/amis/080405.

[34] B. Roy, Multicriteria methodology for decision aiding, Nonconvex Optimization and Its Applications, Springer, 1996. doi:10.1007/978-1-4757-2500-1.

[35] B. Roy, Decision-aid and decision-making, in: Readings in Multiple Criteria Decision Aid, Springer, 1990, pp. 17–35. doi:10.1007/978-3-642-75935-2_2.

[36] J.P. Brans, B. Mareschal, PROMETHEE methods, in: Multiple criteria decision analysis: state of the art surveys, Springer, 2005, pp. 163–186. doi:10.1007/0-387-23081-5_5.

[37] J.C. Fodor, M.R. Roubens, Fuzzy preference modelling and multicriteria decision support, Kluwer Academic Publishers, Dordrecht, 1994.

[38] E. Fernandez, J. Navarro, A new approach to multi-criteria sorting based on fuzzy outranking relations: The THESEUS method, European Journal of Operational Research 213 (2011) 405–413. doi:10.1016/j.ejor.2011.03.036.

[39] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2002), 182–197. doi:10.1109/4235.996017.

[40] E. Fernandez, J. Navarro, A. Duarte, G. Ibarra, Core: A decision support system for regional competitiveness analysis based on multi-criteria sorting, Decision Support Systems 54(3) (2013) 1417–1426. doi:10.1016/j.dss.2012.12.009.

Appendix A

List of acronyms:

DM	Decision Maker
EA	Evolutionary Algorithm
ELECTRE	ELimination Et Choix Traduisant la REalité (Elimination and Choice Expressing the Reality)
H-MCSGA	Hybrid Multi-Criteria Sorting Genetic Algorithm
MOEA	Multi-Objective Evolutionary Algorithm
MOP	Multi-objective Optimization Problem
NO-ACO	Non Outranked Ant Colony Optimization
NSGA-II	Non-dominated Sorting Genetic Algorithm-II
PPPP	Public Project Portfolio Problem
PROMETHEE	Preference Ranking Organization Method for Enrichment Evaluation
ROI	Region of Interest

Notations

Notation	*Description*
Preferential model	
(x, y)	Pair of solutions
$\sigma(x, y)$	Degree of truth of the statement 'x is at least as good as y'
λ	Threshold of acceptable credibility
β	Asymmetry parameter
ε	Symmetry parameter
xPy	Strict preference
xIy	Indifference relation
xQy	Weak preference
xRy	Incomparability relation
xKy	k-Preference
O	Set of feasible solutions in the objective space
S	Set of solutions that strictly outrank x
NS	Non-strictly-outranked frontier
W	Non-strictly-outranked solutions that weakly outrank x
NW	Non-weakly-outranked frontier
$F_n(x)$	Net flow score of x
F	Non-strictly-outranked solutions whose net flow score is greater than the x's
NF	Net-flow non-outranked frontier
THESEUS method	
Ct	Finite set of ordered categories
M	Total number of categories
C_M	Preferred category
U	Universe of objects x
N	Total number of criteria
G	Coherent set of N criteria
T	Set of reference objects or training examples

b_{kh}	Elements of T
C_k	Denotes the category of b_{kh}
$C(x)$	Denotes a potential category assignment of object x
$\succsim$	Denotes the statement 'is not worse than' on the set of categories
D_P	Set of P-inconsistencies
D_Q	Set of Q-inconsistencies
D_I	Set of I-inconsistencies
D_{1I}	Set of first-order I-inconsistencies
D_{2I}	Set of second-order I-inconsistencies
n_P	Cardinality of the set D_P
n_Q	Cardinality of the set D_Q
n_{1I}	Cardinality of the set D_{1I}
n_{2I}	Cardinality of the set D_{2I}
N_1	First objective function of the THESEUS assignment rule
N_2	Second objective function of the THESEUS assignment rule
C_H	Highest category
C_L	Lowest category
Public Project Portfolio Problem	
N	Total number of projects
$f(i)=\langle f_1(i), f_2(i),...,f_p(i)\rangle$	p-dimensional vector representing the ith project
$f_j(i)$	Contribution of project i to the jth objective
x_i	Binary variable that identifies whether or not a project i is included in a portfolio
$x = \langle x_1, x_2,..., x_N\rangle$	Representation of a portfolio (subset of projects)
B	Total budget that the organization is willing to invest
c_i	Cost of the project i
a_i	Area of the project i
L_k	Lower limit of the area k
U_k	Upper limit of the area k
$g_i(k)$	Binary variable that identifies whether or not, a project i corresponds to the area k
$z(x)=\langle z_1(x), z_2(x), z_3(x),..., z_p(x)\rangle$	Represents the quality of a portfolio
$z_j(x)$	Contribution of each project x to the jth objective that compose a portfolio
R_F	Feasible region

Laura Cruz received the M.Sc. degree in Computer Science from Instituto Tecnológico y de Estudios Superiores de Monterrey (Mexico), in 1999 and she received the PhD (Computer Science) degree from Centro Nacional de Investigación y Desarrollo Tecnológico (Mexico), in 2004. She is a full time professor at Madero Institute of Technology, Mexico. Her research interests include optimization techniques, complex networks, autonomous agents and algorithm performance explanation. Laura Cruz is a member of the Mexican National System of Researchers and the Mexican Society of Operation Research. Professor Cruz is referee of some international journals in the frame of metaheuristics optimization.

Eduardo Fernandez received the PhD degree in Computer Aided Design of Electronic Circuits, from Poznan University of Technology, 1987. He is currently Senior Professor of the Faculty of Civil Engineering, Autonomous University of Sinaloa, Mexico. His main areas of interest are multi-criteria and intelligent decision support, multi-objective optimization and group decision models. In those fields he has published more than eighty papers and book chapters. Professor Fernandez is a member of the Mexican National System of Researchers, the Mexican Society of Operation Research, the International EUREKA Network, the International Society on MCDM and the Euro Working Group on Multi-Criteria Decision.

J. Patricia Sanchez she is a PhD student at Inter-institutional Network of Tijuana-Madero-and-Leon Institutes of Technology. She received the M.Sc. degree in Computer Science from Madero Institute of Technology (Mexico), in 2012. Her research interests are in the areas of evolutionary multi-objective optimization, multi-criteria decision making and project portfolio selection.

Carlos A. Coello Coello received the PhD degree in Computer Science from Tulane University, USA, in 1996. He is currently Professor with Distinction (CINVESTAV-3F Researcher) and Chair of the Computer Science Department of CINVESTAV-IPN, in Mexico City, Mexico. His major research interests are: evolutionary multi-objective optimization and constraint-handling techniques for evolutionary algorithms. Professor Coello has authored and co-authored over 355 publications which currently report over 22,000 citations according to Google Scholar (his h-index is 61). He is currently associate editor of the IEEE Transactions on Evolutionary Computation and serves in the editorial board of 14 other international journals.

Claudia Gomez is a member of the Mexican National System of Researchers and the Mexican Society of Operation Research. She is a full time professor at Madero Institute of Technology, Mexico. She received the PhD degree in Advanced Technology from Instituto Politecnico Nacional (Mexico), in 2009. She received the M.Sc. degree in Computer Science from Instituto Tecnologico de Leon (Mexico), in 2000. Her research interests are optimization techniques, complex network and autonomous agents.