

## EVOLUTIONARY MULTI-OBJECTIVE OPTIMIZATION FOR INFERRING OUTRANKING MODEL'S PARAMETERS UNDER SCARCE REFERENCE INFORMATION AND EFFECTS OF REINFORCED PREFERENCE

Eduardo FERNANDEZ<sup>\*</sup> Jorge NAVARRO<sup>\*\*</sup> Gustavo MAZCORRO<sup>\*\*\*</sup>

**Abstract.** Methods based on fuzzy outranking relations constitute one of the main approaches to multiple criteria decision problems. The use of ELECTRE methods require the elicitation of a large number of parameters (weights and different thresholds); but direct eliciting is often a demanding task for the decision-maker (*DM*). For handling intensity-of-preference effects on concordance levels, a generalized concordance model was proposed by Roy and Slowinski which is more complex than previous outranking models. In this paper, an evolutionary multi-objective-based indirect elicitation of the complete ELECTRE III model-parameter set is proposed. The evolutionary multi-objective inference method is successfully extended to inferring reinforced-preference model parameters. Wide experimental evidence is provided to support the proposal, which performs well even working on small size reference sets.

**Keywords:** Multiple criteria analysis, Fuzzy outranking relations, Parameter inference, Evolutionary algorithms.

### 1. Introduction

Many practical decisions can be modeled by using multi-criteria decision analysis. Multi-criteria methods entail a decision-maker (*DM*) reflecting his/her preferences in a pre-specified mathematical structure. Hence, obtaining preference information from the *DM* and formalizing such information into preferential parameters is a crucial aspect in building a multi-criteria decision model ([4]). The development of these models can be based on direct or indirect elicitation procedures. In the first case, the *DM* must specify preferential parameters through an interactive process guided by a decision analyst ([7]). Usually, the *DMs* reveal difficulties when they are inquired to assign values to parameters whose meanings are barely clear for them. On the other hand, indirect procedures, which compose the so-called preference-disaggregation analysis (*PDA*), use regression-like methods for

---

<sup>\*</sup> Corresponding author, Autonomous University of Sinaloa.

<sup>\*\*</sup> Autonomous University of Sinaloa.

<sup>\*\*\*</sup> National Technical Institute, UPIICSA

inferring a set of parameters  $f$  from a battery of decision examples ([7]). The object of this paper is to derive a fitted model consistent with a decision policy embedded in a set of decision examples. According to Doumpos and Zopounidis in [6], such examples may be provided by:

- a) former decisions made by the *DM*;
- b) decisions on a limited set of fictitious actions for which the *DM* can easily express preferential judgments (decision policy); and
- c) decisions on a subset of actions under consideration, for which the *DM* is comfortable expressing a decision policy.

There are some early works related to the *PDA* paradigm (e.g. [15]). Indeed, the process of assessing criterion weights in value-function and utility models (cf. [14, 23, 28]) may be considered an example of the *PDA* approach. In the framework of Multiple Criteria Decision Aid (*MCD*A), Jacquet-Lagrange and Siskos in [13] pioneered the *UTA* method. According to Greco et al. in [11], *MCD*A approaches based on disaggregation paradigms are of increasing interest because they imply relatively less cognitive effort from the *DM*. The direct eliciting settlement has been recently criticized by Marchant and Pirlot ([16, 24]). In [16], Marchant argues that the only valid preferential input-information is that arising from *DM*'s preferential judgments in pairwise comparisons. Another way is to capture explicit or implicit knowledge or policy from *DM*s via rough sets approach, in which the decision policy is modeled by a set of minimal decision rules (e.g. [12]). Our interest here is restricted to *PDA* in ELECTRE methods, which are one of the most popular multicriteria decision tools (e.g. [10, 17, 25]). In ELECTRE-based models, inferring all the parameters simultaneously requires solving a non-linear programming problem with non-convex constraints, which is usually difficult (cf. [5, 19]). According to Doumpos et al. in [7], the relational form of these models and the veto conditions may make it impossible to infer the model parameters in real-size data sets. Otherwise, in small data sets the non-linear problem may be ill-determined; there are many different parameter settlements that are compatible with preference information, but no mathematical programming technique is able to describe the whole compatible parameter settlement. The problem is particularly difficult when effects of reinforced-preference on the credibility of outranking are considered as in [27]. At present, indirect procedures have not been applied to this enhanced outranking model. Besides, the concordance model is more complex, thus requiring even more cognitive effort from the *DM* when a direct settlement is used.

Two recent papers examine the problem of inferring outranking model parameters by evolutionary techniques, both in the context of multi-criteria sorting problems (cf. [7, 9]). In recent years, evolutionary algorithms have rendered powerful tools for solving difficult problems in a variety of fields, in particular, for the treatment of non-linearity and global optimization in polynomial time (cf. [2]). In [7], Doumpos et al. use a differential evolution algorithm for inferring parameter values in the ELECTRE TRI method. In [9], Fernandez et al. propose an evolutionary multi-objective algorithm for inferring parameters of a fuzzy indifference relation model for multi-criteria sorting purposes. Compared with single-objective optimization, the multi-objective approach is (though more complex) more flexible because it allows a richer modelling of preferences. The solution of the parameter inference problem must satisfy several constraints in the parameter space. The *DM* may be

unable to establish the model parameters, but the *DM* may provide subjective information about criterion importance and parameter value ranges. As constraints, these judgments can reduce the search space and help to obtain more acceptable solutions. As an additional advantage, an evolutionary multi-objective algorithm is capable of generating many good compromise solutions in the associated parameter space. As a result of the evolutionary exploration process, a characterization of the complete set of different model parameter settlements is achieved. This information is then used to obtain a final parameter settlement. The first goal of the present paper is to develop an evolutionary multi-objective method for inferring the whole set of ELECTRE III model parameters. The approach combines the preference information contained in a reference set with inter-criteria and intra-criteria soft information arising from the *DM*. Further, the capacity of the evolutionary search is applied to achieve a better characterization of the set of compatible parameter settlements. As a second goal the approach is extended to achieve a more complex model for handling reinforced preferences, as proposed by Roy and Slowinski in [27].

Fundamental problems arise when there is no previous preference information. In such cases the *DM* should provide the required information in terms of preference judgments (paired-comparisons or sorting decisions). As a third goal the paper explores the manner to obtain preference information from the *DM*, which must be operational. The number of required judgments and the cognitive effort from the *DM* should be kept as small as possible. This issue has not been approached by other papers, which have used a rich information contained in reference sets with hundreds, even thousands, of pairs of objects.

Aside of this introduction, the paper is organized as follows: notations and main assumptions are pointed out in Section 2. The optimization model is outlined in Section 3. The method for inferring ELECTRE III parameters is detailed in Section 4. The proposal by Roy and Slowinski in [27] is briefly outlined in Section 5, in which the evolutionary search is also adapted to the new model. Experimental evidence is given in Section 6 through five illustrative examples. Section 7 contains concluding remarks.

## 2. Assumptions and notations

Let us consider a consistent family of criteria  $G=\{g_1, \dots, g_n\}$  defined on a decision set  $A$ . Let  $\succsim$  be a preference relation defined on a subset of  $A \times A$  such that  $x \succsim y$  means that the *DM* is sufficiently confident with the statement ' $x$  is at least as good as  $y$ '.

**Assumption 1:** The *DM* can provide a reference set  $T \subset A \times A$  composed of action pairs  $(a, b)$  satisfying the following property: For each  $(a, b) \in T$ , one of the two statements below is true:

- i)  $a \succsim b$ ;
- ii)  $\text{not } a \succsim b$ .

Let  $\sigma(x, y)$  be a fuzzy outranking relation defined on  $A$ .  $\sigma(x, y)$  may be interpreted as the degree of credibility of the statement ' $x$  is at least as good as  $y$ '.  $\sigma$  can be calculated with respect to the assumptions imposed by ELECTRE III, PROMETHEE, or by any other outranking inspired approach. We are interested in considering how the  $\sigma$ -image depends on a specific settlement of the model parameters (weights and thresholds). Let us denote by

$P$  the set of model parameters to be inferred. Thus, the assessment of the degree of credibility for ‘ $x$  is at least as good as  $y$ ’ is a function  $\sigma(x, y, P)$ .

**Assumption 2:** The  $DM$  has additional information about criterion importance, symmetry, asymmetry, and acceptable parameter ranges. By using such information, the  $DM$  reveals judgments of acceptability or preferences on different parameter settlements.

Now, we denote by  $P^*$  a specific settlement of model parameters. Let us consider a real value  $\lambda \geq 0.5$  and the following crisp binary relations on  $T$ :

- $(x, y) \in S(\lambda)$  iff  $\sigma(x, y, P^*) \geq \lambda$  ( $\lambda$ -outranking)
- $(x, y) \in P(\lambda)$  iff  $\sigma(x, y, P^*) \geq \lambda \wedge \sigma(y, x, P^*) < 0.5$  ( $\lambda$ -strict preference)
- $(x, y) \in Q(\lambda)$  iff  $\sigma(x, y, P^*) \geq \lambda \wedge 0.5 \leq \sigma(y, x, P^*) < \lambda$  ( $\lambda$ -weak preference)
- $(x, y) \in I(\lambda)$  iff  $\sigma(x, y, P^*) \geq \lambda \wedge \sigma(y, x, P^*) \geq \lambda$  ( $\lambda$ -indifference).
- $(x, y) \in R(\lambda)$  iff  $\sigma(x, y, P^*) < \lambda \wedge \sigma(y, x, P^*) < \lambda$  ( $\lambda$ -incomparability).

### 3. Parameter inference by using a multi-criteria error measure

A perfect consistency preference model-decision policy is reflected by the logical implications:

$\forall (x, y) \in T$

$$\sigma(x, y, P^*) \geq \lambda \Rightarrow x \succeq y \quad (1.a)$$

$$x \succeq y \Rightarrow \sigma(x, y, P^*) \geq \lambda \quad (1.b)$$

0-1 logic in Eq. (1.a) is a simplified model. A non-ideal decision maker may associate a degree of truth less than one to the proposition ‘ $x$  is at least as good as  $y$ ’. Equation (1.a) can be considered a fuzzy implication. For a non-ideal decision maker, his/her perception about ‘ $x$  is at least as good as  $y$ ’ is influenced by his/her beliefs about ‘ $y$  is not at least as good as  $x$ ’. Using the above  $\lambda$ -relations, (1.a) and (1.b) can be transformed into:

$\forall (x, y) \in T$

$$xP(\lambda)y \Rightarrow x \succeq y \quad (2.a)$$

$$xQ(\lambda)y \Rightarrow x \succeq y \quad (2.b)$$

$$xI(\lambda)y \Rightarrow x \succeq y \quad (2.c)$$

$$x \succeq y \Rightarrow xS(\lambda)y \quad (2.d)$$

When the model  $\sigma$  suggests a strict preference favoring  $x$  over  $y$ , the statement ‘ $x$  is at least as good as  $y$ ’ is more credible. The implication (2.a) is more credible than (2.b) or (2.c). Yet some effects of reinforced preferences (cf. [27]) could make  $x \succeq y$  true despite  $\sigma(x, y, P^*) < \lambda$ .

Conditions

- 1)  $(x, y) \in P(\lambda)$  with *not*  $x \succeq y$
- 2)  $(x, y) \in Q(\lambda)$  with *not*  $x \succeq y$
- 3)  $(x, y) \in I(\lambda)$  with *not*  $x \succeq y$
- 4)  $(x, y) \in \succeq$  with *not*  $xS(\lambda)y$

are identified as inconsistencies with  $P(\lambda)$ ,  $Q(\lambda)$ ,  $I(\lambda)$ , and  $\succeq$ , respectively. Such discrepancies can be interpreted as errors, deviations of  $\sigma(x, y, P^*)$  from a good model for

the degree of truth of the predicate ‘ $x$  is at least as good as  $y$ ’. Such inconsistencies may arise from inadequate assessments of some model parameters.

Let us consider the following sets:

$$D_P = \{ (x, y) \in P(\lambda) \text{ with } \text{not } x \succeq y \}$$

$$D_Q = \{ (x, y) \in Q(\lambda) \text{ with } \text{not } x \succeq y \}$$

$$D_I = \{ (x, y) \in I(\lambda) \text{ with } \text{not } x \succeq y \}$$

$$D_{\succeq} = \{ (x, y) \in \succeq \text{ with } \text{not } xS(\lambda)y \}$$

$n_P$ ,  $n_Q$ ,  $n_I$  and  $n_{\succeq}$  denote the respective cardinality of the above sets. Obviously, such values depend on  $P$ .

The model parameters should be inferred from the best compromise solution to the multi-objective optimization problem:

$$\begin{aligned} & \text{Minimize } (n_P, n_Q, n_I, n_{\succeq}) \\ & P \in R_F \end{aligned} \quad (3)$$

where  $R_F$  is a feasible region in the parameter space. This region is determined by constraints that the  $DM$  imposes on the model parameters (Assumption 2). In the remainder of the paper we shall denote by  $(n_P, n_Q, n_I, n_{\succeq})^*$  the best compromise solution to Problem 3 in the objective space.

As in [9], we disregard single objective minimization of some error function or any related criterion. Compared with single-objective optimization, a multi-objective approach is more flexible because it allows modelling preferences on different objectives. The different inconsistency measures do not have the same importance and should not be merged into a single objective; inconsistencies with  $P(\lambda)$  seem to be more relevant. The objective  $n_Q$  seems to be a little more important than  $n_I$  and  $n_{\succeq}$ . However, since it is difficult to model the  $DM$ 's priorities with respect to  $n_Q$ ,  $n_I$  and  $n_{\succeq}$ , we use a posterior modelling of preferences. Still, the complexity of solving (3) suggests the application of evolutionary algorithms. These are particularly convenient to solve multi-objective optimization problems because they render approximations to the Pareto frontier in a single run, instead of performing many single-objective optimization processes such as non-heuristic conventional multi-objective optimization techniques do ([2, 9]). An evolutionary-based solution to Problem 3 for inferring the ELECTRE III model parameters is developed in the next section.

Beyond the mathematical complexities, many  $P^*$  may arise as a pre-map from  $(n_P, n_Q, n_I, n_{\succeq})^*$ . From such  $P^*$  the  $DM$  should select a  $P_{\text{best}}^*$  as ‘the most appropriate settlement’. Indeed, this choice is another selection problem whose solution demands the  $DM$  consider the following:

- How representative is the particular solution regarding the whole distribution of parameter settlements?
- How capable is such a solution of restoring additional preference information not included in the reference set?

- To what extent does such a solution agree with the *DM*'s additional information about criterion importance, symmetry, asymmetry, and acceptable parameter ranges?

These issues will be addressed in subsection 4.4.

## 4. Inference of ELECTRE III parameters by evolutionary multi-objective optimization

### 4.1. Brief outline of ELECTRE III

The ELECTRE methods introduced a binary relation  $S$  that is a model of  $\succsim$ . Proposition  $xSy$  (' $x$  outranks  $y$ ') holds if, and only if, the *DM* has sufficient arguments in favor of ' $x$  is at least as good as  $y$ ' and there are no strong arguments against it. In more formal way, the coalition of the criteria in agreement with that proposition is strong enough, and there is no important coalition discordant with it (cf. [26]). This can be expressed by the following logical equivalence [22]:

$$xSy \Leftrightarrow C(x,y) \wedge \sim D(x,y)$$

where:

- $C(x,y)$  is the predicate about the strength of the concordance coalition; this coalition is composed of two criterion subsets:  
 $C(xSy) = \{g_j \in G \text{ such that } g_j(x) - g_j(y) \geq -q_j\};$   
 $C(yQx) = \{g_j \in G \text{ such that } g_j(y) - p_j \leq g_j(x) - g_j(y) - q_j\}$  ( $Q$  denotes 'weak preference');  
 $p_j$  and  $q_j$  denote the preference and indifference thresholds for criterion  $j$  ( $p_j \geq q_j \geq 0$ ).
- $D(x,y)$  is the predicate about the strength of the discordance coalition  $C(yPx) = \{j \in G \text{ such that } g_j(y) - g_j(x) \geq p_j\}$  ( $P$  denotes 'strict preference')
- $\wedge$  and  $\sim$  are the logical connectives for fuzzy conjunction and fuzzy negation, respectively.

Let  $c(x,y)$  denote the truth degree of predicate  $C(x,y)$ . Using the 'product' operator for conjunction, in ELECTRE III the degree of credibility of  $xSy$  is calculated by:

$$\sigma(x,y) = c(x,y) \cdot Nd(x,y) \quad (4)$$

where  $Nd(x,y)$  denotes the truth degree of the non-discordance predicate.

The concordance index  $c(x,y)$  is defined as follows:

$$c(x,y) = \sum_G k_j c_j(x,y) \quad (5)$$

where:

$k_j$  is the weight of the  $j$ -th criterion ( $k_1 + k_2 + \dots + k_N = 1$ )

$c_j(x,y)$  is the marginal (partial) concordance index for the  $j$ -th criterion. This index is calculated by:

$$c_j(x,y) = \begin{cases} 0 & \text{if } j \in C(yPx) \\ (g_j(x) - g_j(y) + p_j) / (p_j - q_j) & \text{if } j \in C(yQx) \\ 1 & \text{if } j \in C(xSy) \end{cases} \quad (6)$$

The partial discordance is measured in comparison with a veto threshold  $v_j$ , which is the maximum difference  $g_j(y) - g_j(x)$  compatible with  $\sigma(x,y) > 0$ . Following Mousseau and Dias

in [21], we shall use a simplification of the original formulation of the discordance indices in the ELECTRE III method given by:

$$Nd(\mathbf{x}, \mathbf{y}) = \min_{j \in C(\mathbf{y}P\mathbf{x})} [1 - d_j(\mathbf{x}, \mathbf{y})] \quad (7)$$

$$d_j(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{iff } \nabla_j \geq v_j \\ (\nabla_j - u_j) / (v_j - u_j) & \text{iff } u_j < \nabla_j < v_j \\ 0 & \text{iff } \nabla_j \leq u_j \end{cases} \quad (8)$$

where  $\nabla_j = g_j(\mathbf{y}) - g_j(\mathbf{x})$  and  $u_j$  is a discordance (pre-veto) threshold (see Figure 1).

$\sigma(\mathbf{x}, \mathbf{y})$  is calculated by combining Equations 5–8 with Equation 4. So, the following parameters must be specified:

- i) the vector of weights;
- ii) the vector of indifference thresholds;
- iii) the vector of preference thresholds;
- iv) the vector of veto thresholds;
- v) the vector of discordance thresholds (only when the simplification suggested in [21] is used).

Additionally, if a crisp outranking relation is built on  $A$ , then a cutting level  $\lambda^*$  should be specified.

## 4.2 Constraints in Problem 3

As we stated above, the *DM* may reveal subjective information about criterion importance and parameter value ranges. The parameter settlement should correspond with such beliefs. Otherwise, the *DM* would feel disappointed with a decision that contradicts his/her feelings. The most obvious situation concerns ‘weights’. Often, the *DM* has a clear idea about the importance of criteria, though is doubtful regarding weight values. The solution must agree with the order of importance in the *DM*’s mind.

In [8], Fernandez et al. stated that the *DM* should be able to create the following binary relations on  $G$ :

$SI = \{(g_m, g_j) \in G \times G \text{ such that ‘criteria } g_m \text{ and } g_j \text{ have approximately equal importance’}\};$

$LI = \{(g_m, g_j) \in G \times G \text{ such that ‘criterion } g_m \text{ is slightly more important than criterion } g_j’\};$

$MI = \{(g_m, g_j) \in G \times G \text{ such that ‘criterion } g_m \text{ is clearly more important than criterion } g_j’\}.$

$MI$ ,  $LI$  and  $SI$  should hold:

- a. If  $g_m MI g_j$  then  $k_m - k_j \geq \beta$
- b. If  $g_m LI g_j$  then  $\beta > k_m - k_j \geq \beta/2$
- c. If  $g_m SI g_j$  then  $|k_m - k_j| < \beta/2$

(9)

where  $\beta$  is a threshold parameter for strict outranking that is proposed to be settled within the interval  $[1/(2n), 1/n]$  (cf. [8]).

In order to obtain a preference-consistent model, the *DM* may impose other constraints:

$$\text{For } j=1, \dots, n \quad (10)$$

$$0 \leq q_{j \min} \leq q_j \leq q_{j \max}$$

$$p_{j \min} \leq p_j \leq p_{j \max} \quad (q_{j \max} < p_{j \min})$$

$$u_{j \min} \leq u_j \leq u_{j \max} \quad (p_{j \max} < u_{j \min})$$

$$v_{j \min} \leq v_j \leq v_{j \max} \quad (u_{j \max} < v_{j \min})$$

If necessary, the *DM* may impose constraints specifying certain inter-criteria asymmetry conditions. For instance:  $v_j \leq v_l \leq v_i$ ,  $q_l \leq q_j$  or  $p_l \leq p_j$ .

If the *DM* felt that the discordance threshold  $u$  should be near to the middle point of the interval  $[p, v]$ , the constraints:

$$|(v_j + p_j)/2 - u_i| \leq \varepsilon (v_j - p_j) \quad (0 < \varepsilon \ll 1) \quad (i=1, \dots, n) \quad (11)$$

could be added.

The *DM* is not obliged to impose the above full set of constraints; some can be removed, and some, added. The algorithm should be able to handle the constraints required by the *DM* in order to find acceptable solutions according to Assumption 2.

### 4.3 Description of the evolutionary approach for inferring the model parameters

We use the Non-dominated Sorting Genetic Algorithm-II (*NSGA-II*) (cf. [3]). *NSGA-II* is one of the most efficient approaches in the literature on evolutionary multi-objective optimization (cf. [2]). This method ranks every member of a  $K'$ -size population according to individual non-domination levels, applies evolutionary operators to build an offspring population, and combines parent and offspring populations in a new pool of  $2K'$  size. This combined population is sorted into non-dominated classes. The next  $K'$ -size population is obtained by selecting the best individuals of the parent-offspring combined pool. In order to keep diversity, a crowding distance (a density estimator) is associated to every individual.

For the selection of 'parents', *NSGA-II* uses a special kind of binary tournament called 'crowded tournament selection operator'. It works as follows: let  $i, j$  be two randomly selected solutions from the parent population. Solution  $i$  wins the tournament over  $j$  whenever one of the following conditions is true:

- 1) if solution  $i$  has a better rank than  $j$ ;
- 2) if they have the same rank but solution  $i$  has a better crowding distance than  $j$  (that is, the crowding distance associated with  $i$  is greater than the associated to  $j$ ).

Point 1 assures the winner lies on a better non-dominated front. Point 2 solves possible ties between solutions, being on the same front, by deciding according to crowding distances. In this case, the winner resides in a less crowded region.

The pseudocode of *NSGA-II* is given below (cf. [2]):

Generate random population (size  $K'$ )

Evaluate objective values

Generate non-dominated fronts

Assign to these fronts rank based on Pareto dominance

Keep the best front (rank) in the population memory

Generate offspring population

Binary tournament selection

Crossover and mutation

For  $i = 1$  to number of generations



*With parent and offspring population*

*Generate non-dominated fronts*

*Assign to these fronts rank based on Pareto dominance*

*Loop (inside) by adding solutions to next generation*

*Starting from the best front until  $K'$  individuals found*

*Calculate crowding distance between points on each front*

*Update the population memory*

*Select points (elitist) on the better front (with better rank)*

*and which are outside a crowding distance*

*Form next generation*

*Binary tournament selection*

*Crossover and mutation*

*Increment generation index*

*End of loop*

2. Individuals are represented by a string composed of  $5n + 1$  positions as shown in Figure

We use one-point crossover (e.g. [18]).  $2n+1$  possible crossover points are defined on the individual (see Figure 3). Given two parents the specific crossover point is randomly generated. This crossover operator keeps feasibility with respect to constraints 9, 10, 11.

We use uniform mutation (e.g. [18]). The operator for mutation is implemented as follows:

Pick a random integer number  $L \in [1; 4n+2]$

II. If  $L \in [1; n]$ , assign  $j=L$ ; then a random real number  $a \in [q_{jmin}; q_{jmax}]$  is generated. Replace  $q_j$  with  $a$ .

III. If  $L \in [n+1; 2n]$ , assign  $j = 2n - L + 1$ ; then a random real number  $a \in [p_{jmin}; p_{jmax}]$  is generated. Replace  $p_j$  with  $a$ . To enforce restriction (11) a random real number  $b \in B = \{x \in \mathcal{R} \text{ such that } |(v_j + p_j)/2 - x| \leq \varepsilon(v_j - p_j)\}$  is generated. Replace  $u_j$  with  $b$ .

IV. If  $L \in [2n+1; 3n]$ , assign  $j = 3n - L + 1$ ; then a random real number  $b \in B$  is generated; interval  $B$  is defined as in II. Replace  $u_j$  with  $b$ .

V. If  $L \in [3n+1; 4n]$ , assign  $j = 4n - L + 1$ ; then a random real number  $a \in [v_{jmin}; v_{jmax}]$  is generated. Replace  $v_j$  with  $a$ . To enforce restriction (11) a random real number  $b \in \{x \in \mathcal{R} \text{, such that } |(v_j + p_j)/2 - x| \leq \varepsilon(v_j - p_j)\}$  is generated; replace  $u_j$  with  $b$ .

VI. If  $L = 4n+1$ ,  $n$  real numbers  $k_j \in (0; 1)$  are randomly generated. We use the approach proposed in [1].  $n-1$  uniform random numbers are generated in  $(0;1)$ ; further, these are ranked  $0 < a_1 < a_2 \dots < a_{n-1} < 1$ , and the weights are calculated as  $k_i = a_i - a_{i-1}$ . Thus, the normalization condition  $(k_1 + k_2 + \dots + k_n = 1)$  is satisfied and the random weights are uniformly distributed. This particular mutation is considered valid only if the constraints given by (9) hold. Otherwise, the random generation is repeated.

VII. If  $L = 4n+2$ , then a random real number  $a \in [\lambda_{min}; 1]$  ( $\lambda_{min} \geq 0.5$ ) is generated. Replace  $\lambda$  with  $a$ .

Note that the above defined mutation operator also keeps feasibility with respect to constraints 9, 10, 11. The initial population is generated from a feasible individual by reiterative mutations.

In a wide range, the algorithm performance was not very sensitive to the settlement of its parameters. No remarkable differences were found with population size  $K' = 100, 150$ , crossover probability  $p_c = 0.5, 0.8$ , mutation probability  $p_m = 0.05, 0.1$ . The results obtained using 1000 generations were better than those with 200 and 500 generations.

Finally, the parameters of the evolutionary search were set to: *Number of generations* = 1000, *Population size* = 100, *Crossover probability* = 0.8, *Mutation probability* = 0.05.

#### 4.4. Final formalization and discussion

As we shall show through some examples, the above evolutionary methodology is able to find a good compromise solution  $(n_P, n_Q, n_I, n_Z)^*$  to Problem 3 in its objective space. Sometimes, an abundant reference information is available (for instance, when there are many statements  $a \succeq b$  from former decisions made by the *DM* or accepted by him/her). But in other situations, the available reference information could be scarce and the cognitive effort from the *DM* should be kept as small as possible. In case of small size reference sets Problem 3 is ill-determined, and the compromise  $(n_P, n_Q, n_I, n_Z)^*$  usually corresponds to many different points in the parameter space. Let us denote by  $\{P^*\}$  the set of points in the parameter space which are the pre-image of  $(n_P, n_Q, n_I, n_Z)^*$ . Each parameter settlement  $P_i^*$  is compatible with the preference information contained in the reference set under constraints 9–11. However, this does not mean that such compatible solution should be accepted by the *DM* as the correct parameter settlement. The *DM* probably has other beliefs and feelings, not contained in 9–11, which should be satisfied (this will be discussed in detail below). It is thus necessary to choose an element  $P_{\text{best}}^* \in \{P^*\}$  as the final solution of the parameter inference problem. The *DM* may decide between two procedures:

- i) to select a particular  $P_i^* \in \{P^*\}$  according to the *DM*'s own judgment; or
- ii) to use the information provided by their distribution in order to select a more acceptable setting.

Let us discuss ii) thoroughly. There are two random factors that explain the deviation of  $P_i^*$  from more acceptable central points:

- 1) The reference set is a population sample; different reference sets lead to different optimal parameter settlements, although the system of *DM*'s preferences is unique;
- 2) Even under the same reference set, the random nature of the evolutionary algorithm produces different solutions.

Thus,  $P^*$  can be considered a random vector. There is a multivariate probability distribution function  $\Psi(P_m^*, P^*)$  that describes the random behavior of  $P^*$ , being  $P_m^*$  its mean point. Let  $P_{\text{near}}^*$  be the nearest  $P^*$  to  $P_m^*$ .  $P_{\text{near}}^*$  may be considered the most central point of  $\Psi(P_m^*, P^*)$ , and from this view, its most representative point.  $P_{\text{near}}^*$  may coincide with  $P_m^*$  when this is capable of restoring the reference information.

As a final formalization, with scarce preference information we propose the following steps:

- 1) The *DM* provides the preference information;
- 2) The *DM* establishes constraints 9–11;
- 3) Perform many runs of the evolutionary algorithm and obtain a good characterization of  $\{P^*\}$ . Let us denote by  $\{P^*\}_{\text{sample}}$  this set and  $P_m$  its mean point ( $P_m$  is an estimator of  $P_m^*$ );
- 4) Provisional assignment of  $P_{\text{best}}^*$ : Check if  $P_m^*$  belongs to  $\{P^*\}$  ( $P_m^*$  is able to restore the preference information from which  $\{P^*\}$  was derived). In the affirmative case, assign  $P_m^*$  to  $P_{\text{best}}^*$ . Otherwise, find  $P_{\text{near}}^*$  in  $\{P^*\}_{\text{sample}}$  (the nearest  $P^*$  to  $P_m^*$ ), and assign it to  $P_{\text{best}}^*$ ;
- 5) Check if the temporary  $P_{\text{best}}^*$  agrees with the whole system of preferences and beliefs from the *DM*. If possible, check its ability to restore some additional preference information not contained in the reference set from which  $\{P^*\}$  was derived. If these conditions are satisfactorily held, the current  $P_{\text{best}}^*$  can be accepted as the final solution of the inference problem.

Alternatively, the *DM* may:

- in 3) separate  $T$  in different subsets, thus obtaining several candidates to be  $P_{\text{best}}^*$  as in 4), which will be judged according to the above conditions (fifth step); or
- in 4) select another element from  $\{P^*\}_{\text{sample}}$ ; or
- modify/add some constraints thus including additional information and repeat from the third step; or
- use  $\{P^*\}_{\text{sample}}$  as starting basis for further refinement in an interactive

*DM*-analyst decision support procedure as was proposed in [7].

In case of abundant reference information, the procedure is reduced to:

- 1'. The *DM* provides the preference information;
- 2'. The *DM* establishes constraints 9–11;

3'. The best compromise  $(n_p, n_Q, n_l, n_z)^*$  and its pre-map  $P_{\text{best}}^*$  are determined.  $P_{\text{best}}^*$  should agree with the whole system of preferences and beliefs from the *DM*. If possible, check its ability to restore some additional preference information not contained in the reference set from which  $(n_p, n_Q, n_l, n_z)^*$  was derived. If these conditions are satisfactorily held,  $P_{\text{best}}^*$  can be accepted as the final solution of the inference problem.

## 5. Handling reinforced preference on the credibility of outranking

In ELECTRE III, the outranking degree of credibility does not include effects of reinforced preference. To illustrate this issue and its role in our proposal, consider two actions  $x, y$  described by three criteria of similar importance taking their values in the interval  $[1, 7]$  (increasing preferences);  $x=(1,1,7)$ ,  $y=(3,3,1)$ . Let us suppose the following parameter settlement: weights  $k_1=k_2=k_3=1/3$ ; veto thresholds  $v_1=v_2=v_3=5$ ; strict preference thresholds  $p_1=p_2=p_3=1$ . According to the model of Equations 4-8,  $\sigma(x,y)=1/3$ ,  $\sigma(y,x)=0$ , so *not*  $xS(\lambda)y$  with  $\lambda \geq 0.5$ . However, the *DM* may feel a very important difference in the third criterion, and hence  $x$  may be preferred. In such a case  $x \succsim y \Rightarrow xS(\lambda)y$  is false in very wide range of reasonable parameter settlements. Hence, in the framework of an indirect

ELECTRE III parameter settlement, such inconsistencies do not come from an inadequate parameter settlement.

In [27], Roy and Slowinski proposed a way of handling reinforced-preference effects for calculating  $c(\mathbf{x}, \mathbf{y})$ . As in the ELECTRE III, each criterion has an importance coefficient  $k_j$  (a ‘weight’) that contributes to  $c(\mathbf{x}, \mathbf{y})$  whenever  $g_j \in C(\mathbf{xSy}) \cup C(\mathbf{yQx})$ . Roy and Slowinski ([27]) give to some criteria  $g_j$  the possibility of increasing their relative importance when the difference  $g_j(\mathbf{x}) - g_j(\mathbf{y})$  exceeds a value  $rp_j$  called reinforced preference threshold. When this threshold is crossed, the value  $k_j$ , used to model the relative importance of  $g_j$  in  $c(\mathbf{x}, \mathbf{y})$ , is replaced with  $w_j k_j$ , where  $w_j > 1$  is called reinforcement factor. Let  $C(\mathbf{xRPy})$  denote the set of criteria for which  $g_j(\mathbf{x}) - g_j(\mathbf{y}) > rp_j$ . Then, Roy and Slowinski ([27]) propose calculating  $c(\mathbf{x}, \mathbf{y})$  as:

$$c(\mathbf{x}, \mathbf{y}) = \frac{\sum_{j \in C(\mathbf{xRPy})} w_j k_j + \sum_{j \in C(\mathbf{xSy}) - C(\mathbf{xRPy})} k_j + \sum_{j \in C(\mathbf{yQx})} \varphi_j k_j}{\sum_{j \in C(\mathbf{xRPy})} w_j k_j + \sum_{j \in G - C(\mathbf{xRPy})} k_j} \quad (12)$$

$$\text{where } \varphi_j = \frac{g_j(\mathbf{x}) - (g_j(\mathbf{y}) - p_j)}{p_j - q_j}$$

The degree of credibility of outranking is calculated as in ELECTRE III, but replacing (5) with (12).

The parameters to be inferred are:

- i) the vector of original importance coefficients  $k_j$ ;
- ii) the vector of reinforcement factors  $w_j$ ;
- iii) the vector of reinforced preference thresholds  $rp_j$ ;
- iv) the vector of indifference thresholds;
- v) the vector of preference thresholds;
- vi) the vector of veto thresholds;
- vii) the vector of pre-veto thresholds;
- viii) the cutting level  $\lambda^*$ .

The mathematical problem is still (3) with constraints (9–11) and constraints on  $rp_j$  and  $w_j$ .  $w_j$  must be greater than 1;  $rp_j$  must be greater than  $p_j$  and should be greater than  $u_j$ . So, in addition to 9–11 we consider the following constraints:

$$\begin{aligned} \text{For } j=1 \dots n \\ u_j &\leq rp_j \leq rp_{jmax} \\ 1 &< w_j \leq w_{jmax} \end{aligned} \quad (13)$$

### 5.1. Description of the evolutionary approach for inferring the model parameters

The evolutionary algorithm and further exploration is as detailed in subsections 4.3 and 4.4. Handling reinforced preferences, the individuals are represented by a string composed of  $7n + 1$  positions as shown in Figure 4.

Again we use one-point crossover.  $3n+1$  possible crossover points are defined on the individual (see Figure 5). Given two parents the specific crossover point is randomly generated.

As in Section 4 the operator for mutation is implemented as follows:

Pick a random integer number  $L \in [1; 6n+2]$

I. If  $L \in [1; n]$ , then a random real number  $a \in [q_{Lmin}; q_{Lmax}]$  is generated. Replace  $q_L$  with  $a$ .

II. If  $L \in [n+1; 2n]$ , assign  $j = 2n-L+1$ ; then a random real number  $a \in [p_{jmin}; p_{jmax}]$  is generated. Replace  $p_j$  with  $a$ . To enforce restriction (11) a random real number  $b \in B = \{x \in \mathcal{R} \text{ such that } |(v_j + p_j)/2 - x| \leq \varepsilon (v_j - p_j)\}$  is generated. Replace  $u_j$  with  $b$ .

III. If  $L \in [2n+1; 3n]$ , assign  $j = 3n-L+1$ ; then a random real number  $b \in B$  is generated; interval  $B$  is defined as in II. Replace  $u_j$  with  $b$ .

IV. If  $L \in [3n+1; 4n]$ , assign  $j = 4n-L+1$ ; then a random real number  $a \in [v_{jmin}; v_{jmax}]$  is generated. Replace  $v_j$  with  $a$ . To enforce restriction (11) a random real number  $b \in \{x \in \mathcal{R} \text{ , such that } |(v_j + p_j)/2 - x| \leq \varepsilon (v_j - p_j)\}$  is generated; replace  $u_j$  with  $b$ .

V. If  $L \in [4n+1; 5n]$ , assign  $j = 5n-L+1$ ; then a random real number  $a \in [rp_{jmin}; rp_{jmax}]$  and  $a \geq u_j$  is generated.

VI. If  $L \in [5n+1; 6n]$ , assign  $j = 6n-L+1$ ; then a random real number  $a \in [w_{jmin}; w_{jmax}]$  is generated.

VII. If  $L = 6n+1$ ,  $n$  real numbers  $k_j \in (0; 1)$  are randomly generated. As in subsection 4.3 we use the approach proposed in [1].  $n-1$  uniform random numbers are generated in  $(0;1)$ ; further, these are ranked  $0 < a_1 < a_2 \dots < a_{n-1} < 1$ , and the weights are calculated as  $k_i = a_i - a_{i-1}$ . This particular mutation is considered valid only if the constraints given by (9) hold. Otherwise, the random generation is repeated.

VIII. If  $L = 6n+2$ , then a random real number  $a \in [\lambda_{min}; 1]$  ( $\lambda_{min} \geq 0.5$ ) is generated. Replace  $\lambda$  with  $a$ .

Note that the above defined genetic operators keep feasibility with respect to constraints 9, 10, 11, 13. The initial population is generated from a feasible individual by reiterative mutations.

In a wide range, the algorithm performance was not very sensitive to the parameter settlement. Finally, the parameters of the evolutionary search were settled as in subsection 4.3.

## 6. Some illustrative examples

### 6.1. First example: Inferring the ELECTRE III model parameters

Sometimes the *DM* can use information from past decisions, based upon personal experiences or any other agent's. The *DM* may not be inquired to provide new preferential information since a set of reference pairs, with high enough cardinality, is available. Nevertheless, when prior information is not available the *DM* must provide preferential information in order to compose a set of reference pairs. To be practical a method for inferring parameters must be capable of deriving good results from a moderate size set of reference pairs, so the *DM's* efforts can be exerted within manageable limits. The capacity of the proposed method in regard to this issue is examined below.

Let us consider the universe  $U$  composed of objects  $(F_1, F_2, F_3, F_4) \in \mathfrak{R}^4$  with  $F_i$  taking real values in  $[1, 7]$  and the symmetric multi-criteria functional model:

$$V = 0.2 (F_1 + F_2 + F_3 + F_4 + (F_1 \cdot F_2 \cdot F_3 \cdot F_4)^{0.25}) \quad (14)$$

Function  $V$  takes on values from  $[1, 7]$  (increasing preference). In order to build the reference set and the preference information on this set, the decision policy will be simulated by:

$$\forall (x, y) \in T \quad x \succsim y \Leftrightarrow V(x) \geq V(y) - 0.25 \quad (15)$$

where 0.25 simulates certain preference threshold.

Note that the model  $V$  is symmetric, and the differences  $|V(4,4,4,4) - V(5,5,5,1)|$ ,  $|V(5,5,5,5) - V(6,6,6,2)|$ ,  $|V(6,6,6,6) - V(7,7,7,3)|$  lie within the threshold 0.25. That is, improvements in three criteria are compensated by a degradation of three units in a single criterion. This suggests that  $v_j$  may not be very different from 3. So, we accept the constraints:

$$\begin{aligned} |k_m - k_j| &< 0.125 \quad m=1, 2, 3; j > m \quad (\text{from 9}) \\ \text{For } j=1, \dots, 4 & \quad (\text{from 10}) \\ 0 &< q_j \leq 0.3 \\ 0.5 &\leq p_j \leq 0.9 \\ 1.0 &\leq u_j \leq 2.4 \\ 2.5 &\leq v_j \leq 6 \\ |(v_j + p_j)/2 - u_i| &\leq 0.1 \quad (v_j - p_j) \quad (\text{from 11}) \end{aligned} \quad (16)$$

Here, we simulate cases in which decisions are made for a limited set of actions for which the  $DM$  can easily express preferential judgments. Thus, only small-size reference sets are considered. Let us suppose that the  $DM$  is capable of revealing preference judgments (simulated by Equation 15) on a small set  $D$  composed of eight actions. In total, 56 judgments such as  $x \succsim y$  or  $\text{not } x \succsim y$  can be made from the elements in  $D$ .

We attempt to minimize the cognitive effort from the  $DM$  making more effective the use of his/her preference information. If we take subsets of six objects, the original 56-elements set is partitioned into twenty eight 30-element subsets, which are composed of the same original preference information. Thus, without additional cognitive effort from the  $DM$ , we can multiply the number of experiments, making the evolutionary search more exhaustive, and reducing the influence of the random character of the original set of eight objects.

The following experiment was repeated three times:

Basic experiment:

- 1) eight actions are randomly generated;
- 2) these actions are separated in groups composed of six actions; 28 such groups are formed; 30 judgments  $x \succsim y$  or  $\text{not } x \succsim y$  are associated to each group;
- 3) for each 30-judgment group, the algorithm of subsection 4.3 is applied with 10 replications;
- 4) for each 30-judgments group the most central point of the obtained Pareto set is found; this is a potential  $P_{\text{best}}^*$ ;
- 5) the most central point of the whole distribution  $\{P^*\}_{\text{sample}}$  is calculated.

For each experiment two particular points are considered: i)  $P_{\text{best}}^*(1)$ , the most central point of the whole distribution (step 5 of subsection 4.4); ii)  $P_{\text{best}}^*(2)$ , the solution of step 4

which is most compatible with the model symmetric form. The results of the first experiment are shown in Table 1. In this table the most symmetric solution taken as  $\mathbf{P}_{\text{best}}^*(2)$  corresponds to row 18<sup>th</sup>. If a fictitious decision maker were represented by the model based on  $V$ , he/she might feel slightly more comfortable accepting  $\mathbf{P}_{\text{best}}^*(2)$ . In Table 2 standard deviations calculated on the non-dominated set in the parameter space are provided.

$\mathbf{P}_{\text{best}}^*(2)$  agrees better with the whole system of preferences and beliefs from the  $DM$ , but  $\mathbf{P}_{\text{best}}^*(1)$  may be judged as more representative of the whole distribution of solutions. Let us check their ability to restore some additional preference information not contained in the original reference set (step 5, subsection 4.4). Let  $f_1$  be the number of inconsistencies with  $\mathbf{x} S(\lambda)\mathbf{y} \Rightarrow \mathbf{x} \succsim \mathbf{y}$ . Let  $f_2$  be the number of inconsistencies with  $\mathbf{x} \succsim \mathbf{y} \Rightarrow \mathbf{x} S(\lambda)\mathbf{y}$ . Assume a sample set  $C$ . The set  $C$  is a sample from the universe. We worked with  $C$ ,  $C'$  and  $C''$ , each composed of 100 random actions and 9900 pairs  $(\mathbf{x}, \mathbf{y})$ . Considering  $C$  the results were:

For  $\mathbf{P}_{\text{best}}^*(1)$ ,  $f_1 = 36$ ,  $f_2 = 3013$ ; for  $\mathbf{P}_{\text{best}}^*(2)$ ,  $f_1 = 43$ ,  $f_2 = 2823$ ;

Due to the effects of intensity of preferences and incomparability, we have many times  $V(\mathbf{x}) > V(\mathbf{y}) - 0.25$  and, however, *not*  $\mathbf{x} S(\lambda)\mathbf{y}$  still holds. Hence,  $f_2$  is increased. This does not mean an inappropriate parameter settlement; rather, this means that ELECTRE-III models cannot fit very well preference function models such as  $V$ . The results for  $C'$  and  $C''$  are shown in Table 3. Both performances are similar.  $\mathbf{P}_{\text{best}}^*(2)$  is slightly in closer agreement with the symmetry of function  $V$ .

Results of the second and third experiments are pointed out in Table 4.

Note that the results for  $\mathbf{P}_{\text{best}}^*(2)$  were very similar in three different experiments.

## 6.2 Second example: Inferring reinforced preference model parameters

Here, the preference information comes from the same model of Equations 14 and 15, but the outranking model handles reinforced preference on the concordance degree as was described in Section 5.

Taking (16) as base the following constraints were imposed:

$$|k_m - k_j| < 0.125 \quad m=1, 2, 3; j > m \quad (17)$$

$$\text{For } j=1, \dots, 4 \quad (18)$$

$$0 < q_j \leq 0.3$$

$$0.5 \leq p_j \leq 0.9$$

$$1.0 \leq u_j \leq 2.4$$

$$2.5 \leq v_j \leq 6$$

$$1 \leq w_j$$

$$2.5 \leq r p_j \leq 6$$

$$|(v_j + p_j)/2 - u_i| \leq 0.1 \quad (v_j - p_j)$$

The following experiment was performed twice. As in subsection 6.1 eight actions are generated at random. These eight actions are divided into groups composed of six actions, so 28 such groups are formed, and 30 judgments  $\mathbf{x} \succsim \mathbf{y}$  or *not*  $\mathbf{x} \succsim \mathbf{y}$  are associated with each group. For every 30-judgment group, a central point  $\mathbf{P}_{\text{best}}^*$  is obtained for each group distribution. Finally, the most central point for the whole distribution is selected. Like before,  $\mathbf{P}_{\text{best}}^*(1)$  denotes the most central point of the whole distribution (step 5 of 4.4); besides,  $\mathbf{P}_{\text{best}}^*(2)$  is the solution of step 4 which is most compatible with the model

symmetric form. Nevertheless,  $P_{\text{best}}^*(1)$  also satisfies symmetry considerations. Some results of the first experiment are provided in Table 5. In this table the most symmetric solution taken as  $P_{\text{best}}^*(2)$  corresponds to the first row. The most central point of the whole distribution of non-dominated points is shown in the last row.

As in 6.1, the appropriateness of  $S(\lambda)$  is measured by the fulfilment of  $x \succeq y \Leftrightarrow xS(\lambda)y$ . Like before, we shall consider a sample set  $C$  and functions  $f_1, f_2$ . The elements in  $C$  are sampled from the universe. We work with  $C, C'$  and  $C''$ , each composed of 100 random actions and 9900 pairs  $(x, y)$ . Considering  $C$ , the results are: for  $P_{\text{best}}^*(1), f_1 = 46, f_2 = 1558$ . For  $P_{\text{best}}^*(2), f_1 = 46, f_2 = 1558$ .

Due to incomparability, we have many times  $V(x) > V(y) - 0.25$ , still, *not*  $xS(\lambda)y$  holds.

The comparison using other random sets is given in Table 6. Results of the second experiment are presented in Table 7.

#### Partial conclusions:

- The results for  $P_{\text{best}}^*(2)$  are very robust. Only negligible changes can be perceived in different solutions. Comparing results of Examples 1 and 2, although coming from different outranking models, the common components of  $P_{\text{best}}^*(2)$  ( $q, p, u, v, k, \lambda^*$ ) are almost identical.
- Regarding the appropriateness measures  $f_1, f_2$  there are no significant differences between  $P_{\text{best}}^*(2)$  and  $P_{\text{best}}^*(1)$ ; the first may be in more agreement with symmetry considerations.
- Regarding  $f_2$  the model of Section 5 seems to perform better than ELECTRE III.

### 6.3 The preference information comes from an outranking model

#### 6.3.1 A four criteria problem

As before, the universe  $U$  is composed of objects  $(g_1, g_2, g_3, g_4) \in \mathfrak{R}^4$  with  $g_i$  taking real values in  $[1, 7]$ . Suppose that  $\sigma(x, y)$  is calculated according to the model of Section 5, with the following parameters:

$$\begin{aligned} q &= (0.2, 0.2, 0.2, 0.2) \\ p &= (0.8, 0.8, 0.8, 0.8) \\ u &= (2.0, 2.0, 2.0, 2.0) \\ v &= (3.0, 3.0, 3.0, 3.0) \\ rp &= (3.0, 3.0, 3.0, 3.0) \\ w &= (2.0, 2.0, 2.0, 2.0) \\ k &= (0.25, 0.25, 0.25, 0.25) \end{aligned} \tag{19}$$

For every  $(x, y) \in U \times U$  we assume that ' $x$  is at least as good as  $y$ ' ( $x \succeq y$ ) if, and only if,  $\sigma(x, y) \geq 0.67$ . The experiment is similar to examples in 6.1 and 6.2. Eight actions are randomly generated and separated into 28 different groups. For each group the algorithm of subsection 4.3 is applied with 10 replications under constraints 17-18, and a  $P_{\text{best}}^*$  candidate is found. Like before  $P_{\text{best}}^*(1)$  and  $P_{\text{best}}^*(2)$  are determined. Results of the first experiment are shown in Table 8. In this table the most symmetric solution taken as  $P_{\text{best}}^*(2)$  corresponds to the fourth row. The most central point ( $P_{\text{best}}^*(1)$ ) is pointed out in the last row. An optimal ideal solution (0,0,0,0) was found in the objective space, to which correspond many solutions in the parameter space. Some standard deviations are provided in Table 9.



The results of model evaluation on three sets, each composed of 100 random actions, are shown in Table 10.

The above experiment was replicated from other eight random elements. Some results are given in Table 11.

From this example we can conclude that:

- 1) The true values of model parameters were closely approached. Even being slightly different from the true values, the inferred model achieved an outstanding reflection of the simulated preferences;
- 2)  $P_{\text{best}}^*(2)$  seems to be robust and slightly outperforming  $P_{\text{best}}^*(1)$ .

### 6.3.2 A ten criteria problem

This experiment generalizes 6.3.1 in order to test robustness with respect to the number of criteria. Now, the universe  $U$  is composed of objects  $(g_1, g_2, \dots, g_{10}) \in \mathfrak{R}^{10}$  with  $g_i$  taking real values in  $[1, 7]$ .  $\sigma(x, y)$  is calculated according to the model of Section 5, with ten-dimensional parameter vectors given by (20):

$$\begin{aligned}
 q &= (0.2, 0.2, \dots, 0.2) \\
 p &= (0.8, 0.8, \dots, 0.8) \\
 u &= (2.0, 2.0, \dots, 2.0) \\
 v &= (3.0, 3.0, \dots, 3.0) \\
 rp &= (3.0, 3.0, \dots, 3.0) \\
 w &= (2.0, 2.0, \dots, 2.0) \\
 k &= (0.25, 0.25, \dots, 0.25)
 \end{aligned} \tag{20}$$

The set of constraints is given by:

$$|k_m - k_j| < 0.125 \quad m=1, \dots, 9; j > m \tag{21}$$

$$\text{For } j=1, \dots, 10 \tag{22}$$

$$0 < q_j \leq 0.3$$

$$0.5 \leq p_j \leq 0.9$$

$$1.0 \leq u_j \leq 2.4$$

$$2.5 \leq v_j \leq 6$$

$$1 \leq w_j$$

$$2.5 \leq rp_j \leq 6$$

$$|(v_j + p_j)/2 - u_i| \leq 0.1 \quad (v_j - p_j)$$

Some results are given in Table 12. The most central point  $P_{\text{best}}^*(1)$  and the most symmetric point  $P_{\text{best}}^*(2)$  are shown in Table 13. Statistic variability calculated on the non-dominated points gives similar results to the previous example. The results of model evaluation on three random sets are presented in Table 14. No significant differences between  $P_{\text{best}}^*(1)$  and  $P_{\text{best}}^*(2)$  are found.

### 6.4 The preference information comes from a real DM

Let us consider the R&D project evaluation problem analyzed in [9]. In such example 81 projects from a set  $B$  were evaluated by a real decision-maker on four criteria; the assignments are shown in Table 15.

The  $g$  criteria were assumed functions with domain  $[1, 7]$ , although only integer values were considered. The  $DM$  stated that: i) full symmetry should exist in the criterion set; ii) there is a one-dimensional strict preference when  $|g_i(\mathbf{x}) - g_j(\mathbf{y})| = 1$  iii) there is no effect of intensity of preference when  $|g_i(\mathbf{x}) - g_j(\mathbf{y})| \leq 2$ ; iv) there are remarkable effects of intensity of preferences when  $|g_i(\mathbf{x}) - g_j(\mathbf{y})| \approx 3$ ; v) the discordance threshold  $u_i$  should not be far from the middle point between strict preference and veto thresholds. In consequence, the  $DM$  imposed the following constraints:

$$|k_m - k_j| < 0.125 \quad m=1, 2, 3; j > m \quad (23)$$

$$\text{For } j=1, \dots, 4 \quad (24)$$

$$0 < q_j < 0.5$$

$$0.5 \leq p_j \leq 1$$

$$1.9 \leq u_j < 2.5$$

$$2.5 \leq v_j \leq 5$$

$$1 \leq w_j$$

$$2.5 \leq rp_j \leq 5$$

$$|(v_j + p_j)/2 - u_i| \leq 0.1 \quad (v_j - p_j)$$

A subset  $D$  composed of 71 projects was randomly chosen from the original set  $B$ . The reference set  $T$  was formed by all the pairs  $(a, b) \in D \times D$  ( $a \neq b$ ). We consider:

$$ab \Leftrightarrow \text{'the assignment of object } a \text{ is at least as good as the assignment of } b' \quad (25)$$

After 20 runs of the evolutionary algorithm described in subsection 5.1, we obtained 7 non-dominated solutions in the objective space, which correspond to 841 different points in the parameter space. Several representative solutions are given in Table 16. High values of the fourth objective are consequence of incomparability, which is not taken into account by the equivalence (25). Regarding the values of the most important objectives and symmetry considerations, we chose as  $P_{\text{best}}^*$  the solution shown in the first row of Table 16.

Let us check the appropriateness of  $S(\lambda)$  with the selected solution by using the above functions  $f_1$  and  $f_2$ . Let us consider the additional preference information contained in the set of pairs  $C = \{(x, y) \in B \times B, x \text{ or } y \text{ belonging to } (B-D), \text{ and } x \neq y\}$  that was not included in the reference set  $T$ . We obtained  $f_1 = 0$  and  $f_2 = 232$  being  $\text{card } C = 1510$ . Compared with the objective values of Table 16, the chosen  $P_{\text{best}}^*$  seems to be a good result.

## 7. Concluding remarks

Several conditions precede an appropriate settlement of outranking model parameters:

- To be a settlement that minimizes error or inconsistency measures when predictions of the model are compared with the real decision policy; that is, the settlement should have the capacity to restore the reference information;
- To satisfy the  $DM$ 's additional information about criterion importance, symmetry, asymmetry and acceptable parameter ranges; that is, the inferred values and their relationship should be meaningful for the  $DM$ ;
- To perform well when the inferred model is compared with preference statements which are not contained in the reference set. That is, it should have capacity for explaining new decisions.

In this paper, point i) has been approached through evolutionary multi-objective optimization of several inconsistency measures. This yields the following advantages:

- Compared with single-objective optimization, a multi-objective approach is more flexible because it allows preferences on different objectives to be properly modelled. The different inconsistency measures do not have the same importance and should not be integrated into a single objective. For instance, inconsistencies regarding  $P(\lambda)$  seem to be more relevant than other types;
- Evolutionary optimization algorithms allow an easy handling of constraints (point ii);
- Evolutionary methods are more robust with respect to an increment of the model's complexity. They can handle more complex models, strong non-linearity and many parameters. For instance, the parameter setting by indirect procedures based on mathematical programming would hardly work with the model of Section 5, even considering a few criteria;
- Evolutionary multi-objective optimization techniques allow a deep exploration of the set of satisfactory solutions; this is an important issue, because as shown by the examples, there could be many different solutions in the parameter space that satisfy the above Condition i), and the *DM*-analyst should have a wide representation of this set in order to select the most appropriate solution.

This proposal is able to obtain the whole set of ELECTRE III model's parameters, in a compatible way with the *DM*'s subjective information about criterion importance and parameter value ranges. Beyond ELECTRE III, this paper shows the use of the evolutionary methodology combined with the ELECTRE model with reinforced preference. This is the most complex outranking model, so a good performance on it may be a sufficient argument to support the general applicability of the method. Besides, there is no precedent of the application of preference-disaggregation analysis to outranking models with reinforced preference effects.

The manner in which the preference information is obtained from the *DM* is operational. Here, we presented a method which uses judgmental information about only a few objects and their comparisons. The way in which the judgments are required is the most permissive. To know, given two propositions '*x* is at least as good as *y*' and '*x* is not at least as good as *y*', the *DM* should select the one with respect to he/she is sufficiently confident about. This paper gives evidence that such small size and easy of handling information is enough to obtain satisfactory results.

The *NSGA-II* algorithm performed very well in some examples with four and ten criteria, identifying parameter settlements satisfying Conditions i), ii) and iii). In the examples of subsection 6.3 the results agree with the true parameter values of the  $\sigma$ -model within a range justified by statistic variability. The proposal seems to be robust in a wide range of criteria. The above advantages of evolutionary methods were tested in several examples. The algorithm performed very well when the outranking model was enhanced to considering reinforced preference.

In a decision support framework, from the collection of possible parameter-settlements, the coupled *DM*-decision analyst should select the final solution of the parameter inference problem. This selection should be made after considering the following criteria:

- I. representativeness of the particular solution in regard to the whole distribution of parameter settlements;
- II. ability of restoring additional preference information, which was not included in the reference set;

III. agreement with additional *DM* information about criterion importance, symmetry, asymmetry, and acceptable parameter ranges.  
Experimental evidence provided by this research suggests that:

- Convenient results can be obtained when considering small size reference sets whose construction requires minor effort by the *DM*. The different settlements that arise in the evolutionary search, in all the examined cases, contain elements satisfying criteria I, II, and III. This conclusion makes the proposal practical;
- When the reference set is small, criterion I often contradicts II and III. Although the most central point of the whole distribution of non-dominated points may be considered as satisfactory, the evolutionary exploration provided some other solutions which may slightly outperform such a central point.

### Acknowledgements

We acknowledge support from CONACyT project grant 57255.

### References

- [1] Butler, J., Jia, J., Dyer, J., Simulation techniques for the sensitivity analysis of multi-criteria decision models, *European Journal of Operational Research* **103**, 1997, 531-546.
- [2] Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A., *Evolutionary Algorithms for Solving Multi Objective Problems*, Second Edition. Springer, New York, 2007.
- [3] Deb, K., *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Chichester-New York-Weinheim-Brisbane-Singapore-Toronto, 2001.
- [4] Dias, L., Mousseau, V., Figueira, J., Climaco, J., An aggregation / disaggregation approach to obtain robust conclusions with ELECTRE-TRI, *European Journal of Operational Research* **138**, 2, 2002, 332-348.
- [5] Dias, L.C., Mousseau, V., Inferring ELECTRE's veto-related parameters from outranking examples, *European Journal of Operational Research*, **170**, 1, 2006, 172-191.
- [6] Doumpos, M., Zopounidis, C., *Multicriteria Decision Aid Classification Methods*, Kluwer Academic Publishers, Dordrecht-Boston- London, 2002.
- [7] Doumpos, M., Marinakis, Y., Marimaki, M., Zopounidis, C., An evolutionary approach to construction of outranking models for multicriteria classification: The case of ELECTRE TRI method, *European Journal of Operational Research*, **199**, 2, 2009, 496-505.
- [8] Fernandez, E., Navarro, J. and Duarte, A., Multicriteria sorting using a valued preference closeness relation, *European Journal of Operational Research*, **185**, 2, 2008, 673-686.
- [9] Fernandez, E., Navarro, J., Bernal, S., Multicriteria sorting using a valued indifference relation under a preference disaggregation paradigm, *European Journal of Operational Research*, **198**, 2, 2009, 602-609.
- [10] Goletsis, Y., Psarras, J., Samouilidis, J.E., Project Ranking in the Armenian Energy Sector Using a Multicriteria Method for Groups, *Annals of Operations Research*, **120**, 2003, 135-157.

- [11] Greco, S., Mousseau, V., Slowinski, R., Ordinal regression revisited: Multiple criteria ranking with a set of additive value functions, *European Journal of Operational Research*, **191**, 2008, 415-435.
- [12] Greco, S., Matarazzo, B., Slowinski, R., Zanakakis, S., Global investing risk: a case study of knowledge assessment via rough sets, *Annals of Operations Research*, **185**, 2011, 103-138.
- [13] Jacquet-Lagrez, E., Siskos, J., Assessing a set of additive utility functions for multicriteria decision making: The UTA method, *European Journal of Operational Research*, **10**, 2, 1982, 151-164.
- [14] Keeney R., Raiffa H., *Decision with multiple objectives: preferences and value tradeoffs*, Wiley, New York, 1976.
- [15] Mangasarian, O.L., Multisurface method for pattern separation, *IEEE Transactions on Information Theory*, **14**, 6, 1968, 801-807.
- [16] Marchant, T., Debate on How to assign numerical values to different parameters that aim at differentiating the role that the criteria have to play in a comprehensive preference model?, *71 Meeting of the Euro Working Group Multiple Criteria Decision Aiding*, Turin, Italy, 2010.
- [17] Mavrotas, G., Diakoulaki, D., Capros, P., Combined MCDA-IP Approach for Project Selection in the Electricity Market, *Annals of Operations Research*, **120**, 2003, 159-170.
- [18] Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer Verlag, Berlin-Heidelberg-New York, 1996.
- [19] Mousseau, V., Slowinski, R., Inferring an ELECTRE-TRI model from assignment examples, *Journal of Global Optimization*, **12**, 2, 1998, 157-174.
- [20] Mousseau, V., Figueira, J., Naux, J.Ph., Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results, *European Journal of Operational Research*, **130**, 2, 2001, 263-275.
- [21] Mousseau, V., Dias, L.C., Valued outranking relations in ELECTRE providing manageable disaggregation procedures, *European Journal of Operational Research*, **156**, 2, 2004, 467-482.
- [22] Perny, P., Multicriteria filtering methods based on concordance and non-discordance principles, *Annals of Operations Research*, **80**, 1998, 137-165.
- [23] Pekelman, D., Sen, S.K., Mathematical programming models for the determination of attribute weights, *Management Science*, **20**, 1974, 1217-1229.
- [24] Pirlot, M., Debate on How to assign numerical values to different parameters that aim at differentiating the role that the criteria have to play in a comprehensive preference model?, *71 Meeting of the Euro Working Group Multiple Criteria Decision Aiding*, Turin, Italy, 2010.
- [25] Teixeira Almeida, A., Multicriteria Modelling of Repair Contract Based on Utility and ELECTRE I Method with Dependability and Service Quality Criteria, *Annals of Operations Research*, **138**, 2005, 113-126.
- [26] Roy, B., The outranking approach and the foundations of ELECTRE methods, in: Bana e Costa, C.A., (ed.), *Reading in Multiple Criteria Decision Aid*, Springer-Verlag, Berlin, 1990, 155-183.

- [27] Roy, B., Slowinski, R., Handling effects of reinforced preference and counter-veto in credibility of outranking, *European Journal of Operational Research*, **188**, 1, 2008, 185-190.
- [28] Srinivasan, V., Shocker, A.D, Linear programming techniques for multidimensional analysis of preferences, *Psychometrika*, **38**, 3, 1973, 337-396.

*Received April, 2012*

# Figures

Figure 1: Partial discordance index  $d_j(\mathbf{x}, \mathbf{y})$

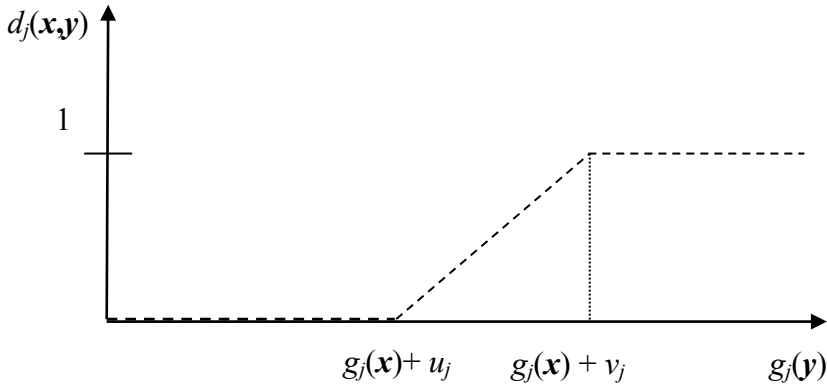


Figure 2: Individual coding

$p_1$	$u_1$	$v_1$	...	...	$p_n$	$u_n$	$v_n$	$k_1$	$k_2$	...	$k_n$	$q_1$	$q_2$	...	$q_n$	$\lambda$
-------	-------	-------	-----	-----	-------	-------	-------	-------	-------	-----	-------	-------	-------	-----	-------	-----------

Figure 3: Possible crossover points

	1	...				n				n+1	n+2	...			2n+1	
$p_1$	$u_1$	$v_1$	...	...	$p_n$	$u_n$	$v_n$	$k_1$	$k_2$	...	$k_n$	$q_1$	$q_2$	...	$q_n$	$\lambda$

Figure 4: Individual coding

$q_1$	...	$q_n$	$p_1$	$u_1$	$v_1$	$rp_1$	...	$p_n$	$u_n$	$v_n$	$rp_n$	$w_1$	...	$w_n$	$k_1$	...	$k_n$	$\lambda$
-------	-----	-------	-------	-------	-------	--------	-----	-------	-------	-------	--------	-------	-----	-------	-------	-----	-------	-----------

Figure 5: Possible crossover point

1		...		n		n+1		...		2n		2n+1		...		3n		3n+1	
$q_1$	...	$q_n$	$p_1$	$u_1$	$v_1$	$rp_1$	...	$p_n$	$u_n$	$v_n$	$rp_n$	$w_1$	...	$w_n$	$k_1$	...	$k_n$	$\lambda$	

Tables  
Table 1. Potential  $P_{\text{best}}^*$  (Example of 6.1, first experiment)

Solution	Optimal ( $n_P, n_Q, n_I, n_Z$ )	$k$	$q$	$p$	$u$	$v$	$\lambda'$
1	0,0,0,0	0,251,0,251,0,250,0,249	0,148,0,150,0,151,0,148	0,698,0,695,0,694,0,695	2,052,2,011,2,042,1,625	3,474,3,443,3,484,2,763	0,700
2	0,0,0,8	0,250,0,251,0,250,0,249	0,152,0,147,0,154,0,154	0,695,0,697,0,694,0,690	2,056,2,041,2,044,2,048	3,505,3,484,3,482,3,488	0,700
3	0,0,0,8	0,250,0,244,0,252,0,254	0,155,0,140,0,140,0,158	0,685,0,682,0,680,0,665	1,928,2,030,1,820,1,605	3,303,3,463,3,101,2,760	0,697
4	0,0,0,8	0,240,0,243,0,241,0,276	0,156,0,155,0,150,0,150	0,551,0,670,0,855,0,631	2,021,2,156,1,804,1,577	3,614,3,716,3,169,2,569	0,702
5	0,0,0,8	0,248,0,253,0,247,0,252	0,143,0,149,0,143,0,158	0,697,0,707,0,665,0,689	1,965,2,015,1,838,1,600	3,329,3,408,3,150,2,739	0,680
6	0,0,0,8	0,248,0,248,0,252,0,253	0,149,0,158,0,155,0,159	0,683,0,672,0,689,0,700	1,829,2,000,1,860,2,022	3,072,3,423,3,123,3,460	0,700
7	0,0,0,8	0,251,0,248,0,248,0,253	0,137,0,156,0,158,0,153	0,687,0,669,0,673,0,679	1,975,2,098,1,851,1,595	3,328,3,613,3,073,2,728	0,700
8	0,0,0,0	0,251,0,249,0,250,0,250	0,153,0,154,0,148,0,150	0,694,0,694,0,694,0,699	2,050,2,053,2,036,2,040	3,468,3,522,3,468,3,475	0,700
9	0,0,0,8	0,252,0,251,0,248,0,249	0,142,0,158,0,151,0,148	0,703,0,695,0,709,0,655	2,052,2,011,2,042,1,625	3,476,3,446,3,484,2,766	0,678
10	0,0,0,8	0,249,0,249,0,250,0,252	0,149,0,153,0,152,0,152	0,705,0,694,0,702,0,666	2,028,2,013,1,997,1,623	3,461,3,387,3,423,2,732	0,700
11	0,0,0,8	0,249,0,252,0,252,0,246	0,157,0,149,0,146,0,149	0,710,0,706,0,699,0,660	2,098,2,035,1,961,1,618	3,574,3,448,3,302,2,765	0,703
12	0,0,0,0	0,250,0,250,0,249,0,251	0,150,0,155,0,154,0,155	0,692,0,692,0,698,0,692	2,036,2,052,2,038,2,044	3,468,3,496,3,487,3,470	0,700
13	0,0,0,8	0,253,0,245,0,252,0,250	0,140,0,149,0,152,0,145	0,710,0,712,0,695,0,678	2,010,2,067,1,994,1,616	3,440,3,486,3,446,2,759	0,701
14	0,0,0,0	0,251,0,251,0,248,0,250	0,150,0,147,0,146,0,149	0,693,0,698,0,698,0,692	2,045,2,039,2,045,2,048	3,481,3,476,3,473,3,484	0,700
15	0,0,0,0	0,250,0,251,0,250,0,249	0,148,0,149,0,149,0,156	0,695,0,695,0,692,0,695	2,042,2,052,2,044,2,037	3,481,3,490,3,474,3,479	0,700
16	0,0,0,0	0,251,0,249,0,251,0,250	0,148,0,149,0,153,0,154	0,698,0,694,0,694,0,691	2,050,2,036,2,048,2,040	3,489,3,473,3,468,3,477	0,700
17	0,0,0,0	0,249,0,251,0,249,0,250	0,152,0,156,0,149,0,153	0,691,0,702,0,699,0,701	2,039,2,050,2,040,2,029	3,477,3,485,3,475,3,445	0,700
18	0,0,0,6	0,250,0,250,0,250,0,250	0,150,0,154,0,147,0,149	0,693,0,694,0,695,0,694	2,036,2,043,2,040,2,036	3,461,3,464,3,461,3,483	0,700
19	0,0,0,8	0,248,0,249,0,249,0,254	0,148,0,152,0,144,0,165	0,682,0,689,0,705,0,663	2,005,2,064,1,841,1,603	3,401,3,554,3,101,2,757	0,705
20	0,0,0,8	0,247,0,256,0,252,0,245	0,156,0,152,0,150,0,154	0,717,0,705,0,657,0,679	1,964,2,27,1,828,1,599	3,248,3,489,3,104,2,750	0,700
21	0,0,0,8	0,250,0,248,0,251,0,250	0,150,0,152,0,150,0,149	0,677,0,694,0,702,0,686	1,834,2,071,1,867,1,978	3,091,3,547,3,141,3,379	0,689
22	0,0,0,8	0,245,0,252,0,249,0,254	0,160,0,145,0,149,0,155	0,701,0,672,0,648,0,649	1,917,2,075,1,824,1,606	3,229,3,509,3,148,2,770	0,700
23	0,0,0,8	0,250,0,250,0,247,0,253	0,159,0,152,0,156,0,147	0,686,0,694,0,710,0,690	2,015,2,038,1,808,1,610	3,402,3,385,3,053,2,738	0,700
24	0,0,0,8	0,248,0,251,0,252,0,250	0,155,0,150,0,156,0,153	0,691,0,694,0,685,0,670	1,810,2,032,1,845,2,004	3,034,3,492,3,104,3,437	0,687
25	0,0,0,8	0,253,0,243,0,248,0,256	0,159,0,151,0,160,0,140	0,675,0,705,0,684,0,620	1,923,2,067,1,885,1,613	3,240,3,583,3,216,2,832	0,700
26	0,0,0,0	0,249,0,248,0,250,0,253	0,155,0,154,0,150,0,154	0,674,0,694,0,694,0,703	1,827,2,064,1,856,1,984	3,092,3,564,3,126,3,385	0,704
27	0,0,0,8	0,252,0,248,0,250,0,249	0,159,0,159,0,148,0,152	0,726,0,693,0,694,0,642	1,987,1,983,1,885,1,605	3,365,3,375,3,136,2,744	0,700
28	0,0,0,8	0,252,0,251,0,247,0,250	0,150,0,152,0,148,0,152	0,687,0,693,0,690,0,686	1,832,2,040,1,840,2,003	3,084,3,484,3,084,3,426	0,700



Table 2. Standard deviations from the example of 6.1

Subset	$\Delta k$	$\Delta q$	$\Delta p$	$\Delta u$	$\Delta v$	$\Delta l$
1	0.025,0.024,0.023,0.021	0.088,0.086,0.088,0.093	0.120,0.108,0.115,0.106	0.280,0.282,0.263,0.055	0.591,0.554,0.570,0.198	0.005
2	0.022,0.023,0.023,0.023	0.088,0.086,0.086,0.086	0.118,0.117,0.113,0.115	0.264,0.264,0.270,0.261	0.606,0.607,0.632,0.595	0.004
3	0.021,0.023,0.021,0.023	0.091,0.090,0.083,0.092	0.113,0.115,0.113,0.121	0.258,0.243,0.144,0.052	0.636,0.596,0.382,0.208	0.004
4	0.023,0.021,0.021,0.026	0.084,0.086,0.087,0.085	0.114,0.112,0.122,0.116	0.231,0.278,0.144,0.056	0.506,0.586,0.396,0.211	0.003
5	0.022,0.027,0.025,0.023	0.080,0.087,0.091,0.076	0.116,0.113,0.124,0.107	0.261,0.215,0.167,0.049	0.549,0.515,0.374,0.194	0.009
6	0.022,0.024,0.023,0.022	0.085,0.086,0.090,0.087	0.115,0.108,0.121,0.122	0.194,0.286,0.153,0.296	0.441,0.642,0.400,0.635	0.003
7	0.021,0.023,0.025,0.022	0.090,0.082,0.081,0.092	0.114,0.117,0.119,0.103	0.249,0.269,0.126,0.045	0.529,0.600,0.361,0.180	0.003
8	0.022,0.022,0.023,0.023	0.087,0.087,0.087,0.085	0.113,0.115,0.117,0.117	0.266,0.259,0.260,0.264	0.600,0.605,0.600,0.588	0.003
10	0.021,0.024,0.022,0.023	0.085,0.087,0.084,0.087	0.111,0.117,0.114,0.110	0.257,0.262,0.250,0.054	0.577,0.593,0.572,0.168	0.002
11	0.021,0.023,0.024,0.021	0.084,0.085,0.089,0.089	0.108,0.116,0.114,0.100	0.241,0.272,0.260,0.059	0.576,0.583,0.578,0.206	0.004
18	0.023,0.022,0.022,0.023	0.088,0.086,0.087,0.086	0.114,0.116,0.115,0.116	0.265,0.263,0.265,0.262	0.602,0.606,0.620,0.590	0.003
19	0.023,0.023,0.023,0.019	0.087,0.090,0.084,0.085	0.111,0.106,0.122,0.097	0.271,0.267,0.159,0.050	0.575,0.629,0.376,0.181	0.006
21	0.023,0.022,0.023,0.023	0.086,0.090,0.088,0.088	0.116,0.115,0.115,0.117	0.187,0.253,0.160,0.306	0.411,0.631,0.394,0.642	0.002
22	0.022,0.022,0.022,0.023	0.083,0.087,0.088,0.089	0.112,0.116,0.112,0.114	0.180,0.258,0.155,0.284	0.385,0.626,0.383,0.612	0.001

Table 3. Comparison using other random sets (example of 6.1)

Sample set	$p_{\text{best}}^*$	$f_1$	$f_2$
C'	1	38	2874
C'	2	39	2832
C''	1	39	2842
C''	2	40	2801

Table 4. Other results from the example of 6.1

	Second experiment		Third experiment	
	$(n_P, n_Q, n_I, n_Z)^*=(0,0,0,1)$		$(n_P, n_Q, n_I, n_Z)^*=(0,0,0,0)$	
	$p_{\text{best}}^{*(1)}$	$p_{\text{best}}^{*(2)}$	$p_{\text{best}}^{*(1)}$	$p_{\text{best}}^{*(2)}$
$k$	(0.246,0.252,0.248,0.254)	(0.251,0.251,0.250,0.249)	(0.247,0.259,0.244,0.250)	(0.249,0.251,0.249,0.250)
$q$	(0.150,0.176,0.153,0.155)	(0.150,0.150,0.150,0.152)	(0.150,0.196,0.154,0.193)	(0.148,0.147,0.149,0.151)
$p$	(0.676,0.708,0.689,0.695)	(0.694,0.694,0.695,0.692)	(0.699,0.725,0.701,0.718)	(0.698,0.699,0.690,0.693)
$u$	(1.764,2.098,1.982,2.040)	(2.042,2.025,2.053,2.041)	(2.052,2.034,2.049,1.830)	(2.061,2.043,2.032,2.057)
$v$	(3.023,3.581,3.364,3.469)	(3.490,3.459,3.500,3.475)	(3.478,3.427,3.487,3.091)	(3.512,3.477,3.463,3.503)
$\lambda$	0.698	0.700	0.698	0.700

Table 5. Some potential  $P_{\text{best}}^*$  from the example of 6.2

Solution	$(n_P, n_Q, n_I, n_\lambda)$	$k$	$w$	$rp$	$q$	$p$	$u$	$v$	$\lambda$
1	0,0,4	0.25,0.25,0.25,0.25	2.00,1.94,2.00,1.97	3.29,3.34,3.27,3.28	0.15,0.15,0.15,0.15	0.70,0.70,0.70,0.70	2.06,2.04,2.05,2.05	3.51,3.49,3.49,3.47	0.700
2	0,0,5	0.25,0.25,0.25,0.25	1.68,1.92,1.96,1.94	3.50,3.42,3.29,3.28	0.16,0.16,0.14,0.15	0.70,0.60,0.71,0.71	2.05,2.10,2.03,2.08	3.46,3.62,3.44,3.53	0.700
3	0,0,7	0.25,0.25,0.24,0.26	2.10,2.05,1.94,2.02	3.21,3.28,3.40,3.25	0.16,0.24,0.16,0.16	0.60,0.73,0.71,0.71	2.06,2.06,2.26,2.04	3.53,3.46,3.89,3.47	0.705
4	0,0,6	0.25,0.25,0.24,0.26	1.63,1.90,1.94,2.01	3.68,3.38,3.54,3.24	0.15,0.24,0.13,0.16	0.69,0.74,0.69,0.71	2.11,2.00,2.27,2.02	3.68,3.56,3.92,3.36	0.700
5	0,0,6	0.25,0.25,0.25,0.25	1.96,1.97,1.10,1.96	3.28,3.22,3.30,3.24	0.15,0.15,0.16,0.15	0.69,0.69,0.70,0.70	2.03,2.04,2.05,2.04	3.45,3.45,3.47,3.48	0.700
6	0,0,7	0.25,0.25,0.25,0.25	2.04,2.04,2.03,2.02	3.27,3.31,3.27,3.28	0.15,0.15,0.15,0.15	0.69,0.60,0.69,0.60	2.05,2.07,2.03,2.05	3.49,3.52,3.46,3.40	0.700
7	0,0,8	0.26,0.25,0.24,0.25	2.09,1.98,1.90,1.86	3.28,3.28,3.45,3.18	0.16,0.23,0.16,0.17	0.70,0.73,0.70,0.70	2.03,2.06,2.27,2.00	3.48,3.43,3.80,3.40	0.688
8	0,0,6	0.25,0.25,0.24,0.25	1.55,1.88,1.95,2.08	3.40,3.39,3.49,3.10	0.15,0.25,0.16,0.15	0.72,0.72,0.71,0.71	2.09,2.09,2.26,2.03	3.57,3.53,3.83,3.46	0.700
MCP	0,0,6	0.26,0.25,0.25,0.25	1.84,1.99,2.01,1.97	3.39,3.35,3.40,3.27	0.15,0.20,0.16,0.18	0.69,0.72,0.70,0.73	2.06,2.07,2.16,2.06	3.50,3.52,3.69,3.46	0.700

Table 6. Comparison using other random sets (Example of 6.2)

Sample set	$p_{best}^*$	$f_1$	$f_2$
C'	1	53	1888
C'	2	53	1888
C''	1	43	1802
C''	2	43	1802

Table 7. Other results from the example of 6.2

	Second experiment	
	$(n_P, n_Q, n_I, n_Z)^* = (0,0,0,4)$	
	$P_{best}^*(1)$	$P_{best}^*(2)$
$k$	(0.248,0.256,0.237,0.258)	(0.250,0.250,0.250,0.251)
$w$	(2.313,2.352,1.993,2.191)	(2.067,1.923,1.997,2.025)
$rp$	(3.255,3.442,3.376,2.983)	(3.274,3.413,3.298,3.285)
$q$	(0.171,0.180,0.147,0.151)	(0.156,0.147,0.151,0.152)
$p$	(0.726,0.707,0.709,0.684)	(0.687,0.696,0.696,0.700)
$u$	(2.211,2.187,2.232,1.990)	(2.061,2.068,2.038,2.049)
$v$	(3.785,3.782,3.843,3.385)	(3.551,3.531,3.488,3.475)
$\lambda$	0.697	0.700

Table 8. Some potential  $P_{\text{best}}^*$  from the example of 6.3.1

Solution	$k$	$w$	$rp$	$q$	$p$	$u$	$v$	$\lambda$
1	(0.249,0.242,0.257,0.252)	(1.979,1.942,1.956,1.944)	(3.270,3.444,3.306,3.251)	(0.224,0.130,0.141,0.144)	(0.710,0.699,0.709,0.697)	(2.046,2.074,2.132,2.032)	(3.471,3.555,3.671,3.473)	0.702
2	(0.251,0.249,0.250,0.250)	(2.027,1.982,2.063,1.979)	(3.486,3.251,3.200,3.346)	(0.141,0.145,0.138,0.152)	(0.694,0.698,0.692,0.698)	(2.182,2.052,2.025,2.055)	(3.769,3.472,3.459,3.498)	0.700
3	(0.250,0.251,0.250,0.248)	(1.939,2.067,2.096,1.952)	(3.511,3.233,3.195,3.352)	(0.156,0.152,0.134,0.150)	(0.692,0.695,0.688,0.695)	(2.189,2.034,2.026,2.063)	(3.786,3.445,3.445,3.504)	0.700
4	(0.250,0.249,0.252,0.250)	(1.964,1.960,2.015,1.919)	(3.423,3.324,3.164,3.351)	(0.149,0.155,0.140,0.154)	(0.705,0.698,0.692,0.696)	(2.094,2.084,2.029,2.035)	(3.536,3.574,3.480,3.455)	0.700
5	(0.246,0.253,0.246,0.255)	(2.046,2.749,1.985,2.142)	(3.415,2.762,3.199,3.219)	(0.173,0.160,0.153,0.158)	(0.754,0.696,0.695,0.695)	(2.233,2.046,2.068,2.065)	(3.794,3.464,3.542,3.498)	0.700
6	(0.247,0.257,0.243,0.253)	(2.016,2.733,2.180,2.083)	(3.481,2.811,3.300,3.441)	(0.176,0.157,0.143,0.148)	(0.759,0.705,0.729,0.691)	(2.170,2.044,2.026,2.060)	(3.650,3.516,3.435,3.491)	0.691
7	(0.244,0.256,0.243,0.256)	(1.827,2.746,2.048,1.952)	(3.475,2.818,3.338,3.245)	(0.161,0.157,0.149,0.148)	(0.753,0.705,0.724,0.695)	(1.993,2.077,2.090,2.048)	(3.323,3.552,3.505,3.467)	0.700
8	(0.245,0.255,0.246,0.255)	(1.986,2.652,1.960,1.985)	(3.443,2.770,3.238,3.248)	(0.173,0.161,0.167,0.151)	(0.775,0.682,0.731,0.707)	(2.255,2.035,2.030,2.012)	(3.839,3.487,3.418,3.393)	0.700
9	(0.250,0.249,0.252,0.250)	(1.936,1.959,2.069,1.912)	(3.411,3.257,3.305,3.326)	(0.150,0.158,0.137,0.148)	(0.706,0.701,0.692,0.703)	(2.090,2.041,2.043,2.061)	(3.532,3.448,3.480,3.473)	0.686
10	(0.246,0.253,0.246,0.256)	(2.053,2.804,2.146,1.997)	(3.508,2.727,3.309,3.237)	(0.152,0.154,0.161,0.149)	(0.763,0.681,0.720,0.715)	(2.229,2.032,2.046,2.020)	(3.833,3.448,3.447,3.472)	0.700
11	(0.244,0.263,0.238,0.256)	(2.229,2.943,2.127,2.051)	(3.485,2.821,3.229,3.176)	(0.156,0.152,0.158,0.164)	(0.792,0.688,0.720,0.693)	(2.211,2.080,2.054,2.009)	(3.737,3.625,3.471,3.421)	0.700
MCP	(0.246,0.257,0.243,0.254)	(1.980,2.713,2.057,1.986)	(3.455,2.849,3.313,3.294)	(0.164,0.163,0.148,0.151)	(0.764,0.696,0.706,0.696)	(2.143,2.043,2.074,2.044)	(3.614,3.471,3.513,3.476)	0.700

 $(n_P, n_Q, n_I, n_Z)^* = (0, 0, 0, 0)$

Table 9. Standard deviations from the example of 6.3.1

Subset	$\Delta k$	$\Delta w$	$\Delta p$	$\Delta q$	$\Delta p$	$\Delta u$	$\Delta v$	$\Delta$
1	(0.024,0.021,0.021,0.022)	(0.799,0.834,0.810,0.823)	(0.690,0.583,0.689,0.699)	(0.052,0.085,0.083,0.085)	(0.115,0.115,0.118,0.117)	(0.256,0.250,0.216,0.259)	(0.554,0.603,0.566,0.598)	0.005
2	(0.022,0.021,0.023,0.022)	(0.753,0.814,0.799,0.800)	(0.685,0.573,0.701,0.759)	(0.052,0.090,0.090,0.093)	(0.114,0.105,0.107,0.118)	(0.269,0.244,0.168,0.257)	(0.562,0.575,0.495,0.568)	0.006
3	(0.024,0.022,0.022,0.023)	(0.811,0.818,0.814,0.836)	(0.708,0.721,0.809,0.710)	(0.087,0.088,0.087,0.086)	(0.117,0.116,0.115,0.118)	(0.269,0.278,0.175,0.268)	(0.594,0.611,0.390,0.619)	0.004
4	(0.023,0.020,0.022,0.023)	(0.734,0.805,0.816,0.859)	(0.655,0.593,0.783,0.723)	(0.050,0.088,0.089,0.085)	(0.124,0.112,0.123,0.120)	(0.270,0.240,0.162,0.249)	(0.572,0.588,0.377,0.604)	0.005
5	(0.022,0.022,0.023,0.023)	(0.816,0.730,0.800,0.813)	(0.719,0.693,0.699,0.680)	(0.089,0.085,0.087,0.085)	(0.110,0.115,0.109,0.115)	(0.272,0.267,0.265,0.265)	(0.608,0.641,0.616,0.605)	0.006
6	(0.022,0.022,0.022,0.023)	(0.808,0.737,0.828,0.751)	(0.720,0.687,0.701,0.670)	(0.085,0.085,0.085,0.084)	(0.117,0.114,0.115,0.116)	(0.267,0.257,0.215,0.271)	(0.605,0.588,0.500,0.585)	0.006
7	(0.021,0.023,0.023,0.022)	(0.752,0.705,0.810,0.800)	(0.644,0.531,0.713,0.722)	(0.073,0.091,0.088,0.089)	(0.121,0.117,0.116,0.117)	(0.272,0.259,0.164,0.244)	(0.619,0.569,0.445,0.520)	0.006
8	(0.023,0.023,0.024,0.023)	(0.817,0.712,0.825,0.798)	(0.672,0.759,0.806,0.675)	(0.087,0.088,0.082,0.090)	(0.117,0.118,0.111,0.113)	(0.248,0.263,0.166,0.268)	(0.584,0.618,0.386,0.627)	0.001
9	(0.021,0.023,0.022,0.023)	(0.750,0.745,0.791,0.778)	(0.676,0.733,0.794,0.724)	(0.088,0.090,0.084,0.087)	(0.114,0.112,0.101,0.112)	(0.278,0.261,0.114,0.252)	(0.579,0.541,0.337,0.592)	0.001
10	(0.022,0.022,0.023,0.022)	(0.816,0.686,0.828,0.802)	(0.688,0.712,0.770,0.648)	(0.083,0.088,0.083,0.085)	(0.117,0.116,0.113,0.116)	(0.266,0.260,0.170,0.281)	(0.573,0.587,0.417,0.628)	0.002
11	(0.022,0.023,0.023,0.022)	(0.808,0.837,0.811,0.820)	(0.682,0.707,0.788,0.686)	(0.084,0.086,0.085,0.087)	(0.111,0.116,0.115,0.115)	(0.263,0.256,0.166,0.272)	(0.603,0.605,0.391,0.611)	0.001

**Table 10. Comparison using random sets (Example of 6.3.1)**

Sample set	$p_{\text{best}}^*$	$f_1$	$f_2$
C'	1	4	49
C'	2	0	0
C''	1	8	60
C''	2	0	0
C''	1	3	61
C''	2	0	0

**Table 11. Other results from the example of 6.3.1**

	Second experiment	
	$(n_p, n_Q, n_I, n_{\geq})^* = (0,0,0,0)$	
	$p_{\text{best}}^*(1)$	$p_{\text{best}}^*(2)$
$k$	(0.249,0.247,0.253,0.252)	(0.248,0.253,0.249,0.251)
$w$	(2.114,2.152,1.995,2.068)	(2.001,2.305,1.952,2.128)
$rp$	(3.181,3.365,3.202,3.169)	(3.262,3.282,3.275,3.156)
$q$	(0.195,0.150,0.159,0.150)	(0.153,0.148,0.161,0.150)
$p$	(0.712,0.695,0.707,0.695)	(0.699,0.692,0.716,0.697)
$u$	(2.039,2.061,1.984,1.991)	(2.032,2.044,2.046,2.054)
$v$	(3.459,3.519,3.332,3.384)	(3.444,3.489,3.466,3.509)
$\lambda$	0.700	0.700

Table 12. Some potencial  $P_{\text{best}}^*$  from the example of 6.3.2

$k$	[0.100,0.099,0.100,0.100,0.101,0.099,0.100,0.100,0.100,0.101]
$w$	[2.112,1.895,1.893,1.892,2.375,2.257,1.939,2.338,1.917,2.994]
$rp$	[3.132,3.419,3.219,3.353,2.941,3.289,3.278,3.178,3.024,2.662]
$q$	[0.169,0.145,0.199,0.159,0.244,0.151,0.141,0.146,0.122,0.155]
$p$	[0.693,0.724,0.784,0.665,0.735,0.697,0.687,0.709,0.669,0.682]
$u$	[2.045,2.072,2.050,2.168,2.009,2.131,2.066,2.096,1.968,1.981]
$v$	[3.502,3.602,3.401,3.727,3.385,3.696,3.639,3.648,3.385,3.398]
$\lambda$	0.690
$k$	[0.100,0.099,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.101]
$w$	[1.952,1.952,2.202,1.574,2.112,1.840,2.117,2.057,2.046,2.998]
$rp$	[3.292,3.332,3.248,3.404,3.141,3.356,3.290,3.367,3.119,2.717]
$q$	[0.162,0.142,0.204,0.148,0.232,0.140,0.156,0.174,0.153,0.160]
$p$	[0.668,0.665,0.803,0.706,0.712,0.722,0.701,0.706,0.715,0.685]
$u$	[1.974,2.137,2.050,2.195,2.036,2.144,1.983,2.111,1.988,2.007]
$v$	[3.369,3.648,3.343,3.759,3.368,3.585,3.444,3.614,3.432,3.480]
$\lambda$	0.695
$k$	[0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100]
$w$	[1.951,2.013,2.006,2.000,2.162,2.217,2.000,2.359,2.036,2.069]
$rp$	[3.317,3.249,3.285,3.232,3.126,3.065,3.256,3.217,3.214,3.278]
$q$	[0.153,0.147,0.155,0.148,0.147,0.148,0.151,0.146,0.148,0.148]
$p$	[0.698,0.692,0.721,0.696,0.701,0.703,0.692,0.696,0.704,0.691]
$u$	[2.046,2.053,2.019,2.038,2.049,2.040,2.039,2.047,2.033,2.043]
$v$	[3.482,3.488,3.416,3.463,3.477,3.455,3.468,3.475,3.438,3.450]
$\lambda$	0.700
$k$	[0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100]
$w$	[1.944,1.837,2.065,2.025,2.060,1.944,1.661,2.087,2.051,2.998]
$rp$	[3.214,3.147,3.499,3.515,3.183,3.263,3.254,3.272,3.347,2.740]
$q$	[0.136,0.149,0.194,0.152,0.228,0.156,0.156,0.145,0.148,0.133]
$p$	[0.692,0.690,0.810,0.688,0.714,0.673,0.673,0.730,0.703,0.734]
$u$	[1.994,2.051,2.130,2.244,2.010,2.153,2.044,2.004,2.004,1.974]
$v$	[3.382,3.551,3.509,3.880,3.436,3.657,3.379,3.407,3.430,3.323]
$\lambda$	0.699
$k$	[0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100]
$w$	[2.026,1.970,2.000,2.010,2.245,2.216,1.962,2.280,1.971,1.972]
$rp$	[3.304,3.273,3.282,3.315,3.076,3.055,3.322,3.306,3.258,3.365]
$q$	[0.151,0.146,0.155,0.151,0.146,0.145,0.161,0.152,0.152,0.151]
$p$	[0.692,0.693,0.713,0.697,0.702,0.700,0.704,0.707,0.697,0.702]
$u$	[2.022,2.045,2.025,2.052,2.028,2.021,2.058,2.050,2.038,2.039]
$v$	[3.460,3.493,3.407,3.538,3.438,3.454,3.492,3.478,3.458,3.458]
$\lambda$	0.700



**Table 13. Final solutions from the example of 6.3.2**

	$(n_P, n_Q, n_I, n_Z)^* = (0,0,0,0)$	
	$p_{\text{best}}^*(1)$	$p_{\text{best}}^*(2)$
<b>k</b>	(0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100)	(0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100,0.100)
<b>w</b>	(2.033,1.984,2.003,2.002,2.117,2.131,1.958,2.212,2.016,2.431)	(1.982,2.017,1.974,2.009,2.012,2.030,2.003,1.988,2.007,2.013)
<b>rp</b>	(3.278,3.272,3.265,3.330,3.167,3.231,3.276,3.297,3.125,3.052)	(3.242,3.248,3.275,3.284,3.264,3.258,3.282,3.257,3.298,3.309)
<b>q</b>	(0.150,0.150,0.173,0.150,0.185,0.147,0.150,0.152,0.151,0.151)	(0.152,0.150,0.154,0.147,0.144,0.151,0.151,0.145,0.153,0.150)
<b>p</b>	(0.692,0.691,0.750,0.694,0.704,0.695,0.693,0.695,0.697,0.699)	(0.694,0.686,0.700,0.702,0.695,0.699,0.696,0.694,0.693,0.698)
<b>u</b>	(2.039,2.050,2.030,2.107,2.024,2.069,2.040,2.043,1.903,2.026)	(2.040,2.052,2.023,2.050,2.044,2.035,2.046,2.039,2.046,2.052)
<b>v</b>	(3.492,3.493,3.393,3.599,3.430,3.524,3.477,3.474,3.197,3.448)	(3.476,3.497,3.411,3.483,3.497,3.470,3.489,3.473,3.492,3.477)
<b><math>\lambda</math></b>	0.700	0.695

**Table 14. Comparison using random sets (Example of 6.3.2)**

Sample set	$p_{\text{best}}^*$	$f_1$	$f_2$
C'	1	24	139
C'	2	26	137
C''	1	19	125
C''	2	22	122
C''	1	20	111
C''	2	16	113

**Table 15. Reference assignments (example of 6.4)**

#	C1	C2	C3	C4	Global Impact
1	7	4	7	7	Exceptional
2	6	6	6	6	Exceptional
3	4	4	4	4	Very High
4	2	4	4	4	High
5	2	2	4	4	Above Average or High
6	2	2	2	4	Above Average
7	2	2	2	2	Average
8	1	2	2	2	Low or Below Average
9	1	1	1	1	Very Low
10	3	3	3	6	High
11	3	3	6	3	High
12	3	3	6	6	Very High

#	C1	C2	C3	C4	Global Impact
42	2	7	6	3	Very High
43	3	6	2	3	High
44	6	3	1	7	Very High
45	1	1	2	3	Below Average
46	3	2	1	2	Average
47	5	5	6	3	Very High
48	4	5	6	4	Very High
49	4	4	2	5	High
50	3	2	2	2	Above Average
51	6	5	6	1	Very High
52	4	6	2	1	High
53	1	6	2	4	High

13	3	6	3	3	High
14	3	6	3	6	Very High
15	3	6	6	3	Very High
16	3	6	6	6	Very High
17	6	3	3	3	High
18	6	6	3	6	Very High
19	6	3	6	3	Very High
20	6	3	6	6	Very High
21	6	6	3	3	Very High
22	6	3	3	6	Very High
23	6	6	6	3	Very High
24	2	2	5	1	Average
25	5	1	2	2	Average
26	5	5	1	2	High
27	2	5	1	2	Average
28	1	5	1	3	Average
29	3	7	7	7	Exceptional
30	7	7	3	7	Exceptional
31	7	7	7	3	Exceptional
32	5	5	3	1	High
33	7	2	5	3	Very High
34	1	1	4	4	Average
35	1	1	5	1	Average
36	1	3	6	1	High
37	1	1	1	6	Above Average
38	1	1	1	2	Very Low
39	1	1	1	7	High
40	1	1	1	3	Low
41	1	1	1	4	Below Average
54	1	6	3	4	High
55	3	1	4	1	Average
56	5	2	3	6	High
57	6	3	7	2	Very High
58	3	4	2	6	High
59	6	6	7	1	Very High
60	4	2	4	3	High
61	2	1	4	5	Above Average or High
62	5	4	1	3	High
63	6	6	7	7	Exceptional
64	3	4	6	4	High
65	4	3	2	7	Very High
66	5	2	6	6	Very High
67	2	7	3	3	Very High
68	6	5	1	6	Very High
69	2	5	7	3	Very High
70	4	1	4	2	Average
71	4	7	3	1	Very High
72	1	6	5	6	Very High
73	4	3	6	2	High
74	6	1	6	4	Very High
75	3	5	5	1	High
76	2	4	3	5	High
77	5	3	2	2	High
78	1	4	4	6	High
79	6	5	2	3	High
80	2	2	5	2	Above Average or High
81	1	7	5	6	Very High

Table 16. Some non-dominated solutions from the example of 6.4

Sol	$n_w d_w n_t$	$k$	$w$	$rp$	$Q$	$p$	$U$	$v$	$\lambda$
1	0.0,0.814	0.24,0.25,0.26,0.25	3.03,3.07,2.89,3.70	3.43,3.00,2.94,2.60	0.32,0.36,0.16,0.47	0.94,0.87,0.92,0.90	2.01,2.06,2.02,2.15	2.8,2.9,2.8,2.9	0.76
2	0.0,0.814	0.24,0.25,0.26,0.25	3.03,3.07,2.89,3.70	3.43,3.00,2.94,2.60	0.32,0.12,0.02,0.47	0.94,0.87,0.92,0.90	2.01,2.06,2.02,2.15	2.8,2.9,2.8,2.9	0.76
3	0.0,0.814	0.25,0.26,0.25,0.25	3.00,3.00,3.00,3.79	3.00,2.62,3.00,2.56	0.39,0.29,0.19,0.15	0.83,0.92,0.995,0.92	2.05,2.09,2.09,2.06	2.9,2.9,2.9,2.9	0.76
4	0.1,19.409	0.245,0.25,0.25,0.25	1.24,1.00,3.99,3.61	3.00,4.00,2.77,2.65	0.02,0.37,0.23,0.09	0.75,0.65,0.97,0.92	2.04,2.01,2.04,2.12	2.9,2.9,2.8,3.0	0.70
5	1.0,19.409	0.25,0.26,0.26,0.24	1.24,1.00,3.99,3.82	3.00,4.00,2.77,2.65	0.02,0.37,0.23,0.09	0.75,0.65,0.97,0.92	2.03,2.01,2.04,2.12	2.9,2.9,2.8,3.0	0.70
6	2.1,19.313	0.24,0.25,0.26,0.25	1.05,1.00,1.00,3.00	3.13,3.00,3.00,2.83	0.01,0.06,0.03,0.30	0.68,0.99,0.99,0.98	2.03,2.01,2.09,2.04	2.9,2.7,2.9,2.8	0.66
7	3.0,19.313	0.25,0.25,0.26,0.25	1.05,1.00,1.00,3.00	3.13,4.14,3.00,2.83	0.26,0.06,0.03,0.23	0.68,0.99,0.985,0.99	2.03,2.01,2.09,2.00	2.9,2.7,2.9,2.8	0.66
8	5.1,19.298	0.24,0.26,0.25,0.26	1.05,1.00,1.00,3.00	3.13,4.14,3.00,2.83	0.09,0.06,0.03,0.23	0.68,0.99,0.985,0.99	2.03,2.01,2.09,2.04	2.9,2.7,2.9,2.8	0.66
9	6.0,19.298	0.26,0.24,0.24,0.26	1.05,1.00,1.00,3.00	3.13,4.14,3.00,2.83	0.26,0.06,0.03,0.23	0.68,0.99,0.985,0.99	2.03,2.01,2.09,2.00	2.9,2.7,2.9,2.8	0.66