Partial Preference Models Using Translated Cones in Discrete Multi-Objective Optimization

Sami Kaddani*†, Satya Tamby*, Daniel Vanderpooten* and Jean-Michel Vanpeperstraete†
*Université Paris-Dauphine, PSL Research University, CNRS, UMR [7243], LAMSADE
75016 PARIS, FRANCE
Email :{sami.kaddani,satya.tamby,daniel.vanderpooten}@lamsade.dauphine.fr
†DCNS Research
280, avenue Aristide Briand, 92220 Bagneux, France
Email: {sami.kaddani,jean-michel.vanpeperstraete}@dcnsgroup.com

***Abstract*—Decision makers tend to define their optimization problems as multi-objective optimization problems. Generating the whole nondominated set is often unrealistic due to its size but also because most of these points are irrelevant to the decision maker. Another approach consists in obtaining preference information and integrating it a priori, in order to reduce the size of the nondominated set and have a gain in computation time. In this work we focus on a partial preference relation based on requirement and tolerance thresholds that translate the Pareto dominance cone. After introducing this preference relation, we present adaptations to use it in existing discrete multi-objective optimization algorithms. Numerical experiments on multi-objective combinatorial optimization problems show the applicabiliy of our approach.**

I. Introduction

In many real-world optimization problems it is required to take into account several conflicting criteria. In this context the Decision Maker (DM) is interested in efficient solutions, which cannot be improved simultaneously on all objectives. The image of an efficient solution in the objective space is called a nondominated point. Multi-objective optimization (MOO), and especially multi-objective combinatorial optimizaton (MOCO), problems are subject to several limits. The main difficulty is related to the large size of the nondominated set, which is in the worst case exponential in the size of the instance. This often results in prohibitive computation times. Moreover, most of the solutions presented to the DM are actually irrelevant with respect to his/her preferences. To overcome this issue, preference integration in multi-objective optimization has been more and more investigated. Preference information of the DM can be integrated before (a priori), during (interactive) or after (a posteriori) the optimization. Concerning MOCO problems, the a posteriori approach is limited by the computation time. Interactive methods, which favor an exploration of the solutions by iteratively optimizing a scalarizing function with evolving parameters, require the involvement of the DM during all iterations [4]. Therefore these methods cannot be used in all situations.

The a priori approach is usually based on optimizing a scalarizing function (weighted sum, achievement function [14], ordered weighted average [15], Choquet integral [5], etc.), whose parameters are defined before the optimization. These parameters reflect the preferences of the DM. Some authors also focus on preference relations based on these scalarizing functions with partial preference information on the parameters (see, e.g., [1], [3]). Multi-objective optimization with respect to a dominance cone representing the DM's preferences has also been studied (e.g. [6]). In many of these approaches, the DM is asked to defined trade-offs between units of criteria compared to units of other criteria. In this paper, we study a partial preference relation that defines several situations of trade-offs between criteria with an alternative framework.

The preference relation extends Pareto dominance by defining additional cases of dominance with values representing a requirement or a tolerance on the criterion vector. A point dominates another one if, in spite of requirement thresholds on some criteria and thanks to tolerance thresholds on others, it is considered better. This is similar to the type of information required by some interactive procedures, e.g. in [9], where the DM is asked to indicate the objectives to be increased and those to be decreased. This preference relation presents numerous advantages. The trade-offs can easily take account of the notion of non-compensation. Moreover the DM can define several cases where he/she can set requirement and tolerance values to establish dominance, and thus integrate different points of view. The requirement and tolerance thresholds can either be determined as constant or variable. Each threshold being defined on the scale of its corresponding criterion, this approach easily handles heterogeneous criterion scales. Furthermore, the preference relation is not necessarily transitive. Since it adds additional cases of dominance, the number of nondominated points according to this preference relation is lower than the Pareto nondominated set. We present how to apply a priori this preference relation in discrete multi-objective optimization.

Preliminaries are defined in Section II, followed by a presentation of the preference relation in Section III. After explaining the adaptation to a discrete multi-objective optimization algorithm in Section IV, we show the performance of our approach on MOCO problems in Section V. Conclusions and perspectives are provided in a final section.

II. PRELIMINARIES

We recall basic concepts and notations related to multi-objective optimization. Consider a multi-objective optimization problem with $p \geq 2$ criteria where X denotes the set of feasible solutions. Each solution x in X is represented in the criterion space by its corresponding criterion vector $f(x) = (f_1(x), f_2(x), ..., f_p(x))$. We assume in the following that each criterion is to be minimized, and we formulate our problem as follows.

$$\begin{cases} \min & f(x) = (f_1(x), f_2(x), ..., f_p(x)) \\ \text{s.t.} & x \in X \end{cases} \quad (1)$$

Let $Y = f(X)$ denote the set of all feasible points in the objective space. Considering two feasible solutions x and x', and their corresponding feasible points $y = f(x)$ and $y' = f(x')$, the following dominance relation can be defined:

$$y \leqq y' \text{ (y weakly dominates y')} \Leftrightarrow y_i \leq y'_i \text{ for all } i = 1, ..., p$$
$$y \leq y' \text{ (y dominates y')} \Leftrightarrow y \leqq y' \text{ and } y \neq y'$$

We refer to $N(Y)$ as the nondominated set of Y.

$$N(Y) = \{y \in Y : \text{there exists no } y' \in Y \text{ such that } y' \leq y\}$$

Solving problem (1) is understood here as computing $N(Y)$ and providing one efficient solution for each nondominated point.

Typically partial information on the DM's preferences can be represented by a binary preference relation R, where y R-dominates y' is denoted by yRy'. We refer to $N(Y, R)$ as the nondominated set of Y with respect to relation R, simply called *R-nondominated set* of Y.

$$N(Y, R) = \{y \in Y : \text{there exists no } y' \in Y \text{ such that } y'Ry\}$$

III. PRESENTATION OF THE PREFERENCE RELATION

The preference relation defines additional cases of dominance by translating the Pareto dominance cone, using positive (requirement) or negative (tolerance) thresholds on each criterion. After giving the definition of a threshold vector and the general preference relation, we propose an interpretation of this relation and give some particular cases.

A. Definition

The threshold vector can either be defined as constant or variable.

Definition 1. Let y be a point in $\mathbb{R}^p$, $\Delta(y)$ be a threshold vector in $\mathbb{R}^p$ and $g : \mathbb{R}^p \longrightarrow \mathbb{R}^p$ be a function in $\mathbb{R}^p$. Then, we have

$$\Delta_i(y) = g_i(y), i = 1, ..., p$$

For the sake of simplicity, we define $\Delta(y)$ as a linear function of y. We propose a formulation with two parameters a, b as follows.

Definition 2. Let y be a point in $\mathbb{R}^p$, $\Delta(y)$ be a threshold vector in $\mathbb{R}^p$ and a, b be two vectors in $\mathbb{R}^p$. Then, we have

$$\Delta_i(y) = a_i y_i + b_i, i = 1, ..., p$$

If $a_i = 0, i = 1, ..., p$, then the threshold vector $\Delta(y)$ is constant.

Definition 2 includes thresholds being constant or calculated as a percentage of a criterion value.

We give the definition of the preference relation hereafter.

Definition 3. Let y, y' be two points of Y and $\Delta(y)$ be a threshold vector in $\mathbb{R}^p$.

$$yR_\Delta y' \text{ if and only if } y + \Delta(y) \leqq y' \text{ and } y \neq y'.$$

where $y + \Delta(y)$ is the vector of components $y_i + \Delta_i(y), i = 1, ..., p$.

The threshold vector $\Delta(y)$ is always defined by the left component of the pair (y, y'). Therefore, by a slight abuse of notation, the threshold vector will be denoted by Δ from this point.

The following results exhibits an inclusion property between two threshold vectors.

Proposition 4. *Let Δ, Δ' be two threshold vectors in $\mathbb{R}^p$ such that $\Delta \leqq \Delta'$. Then, we have:*

$$N(Y, R_\Delta) \subseteq N(Y, R_{\Delta'})$$

Proof. Let $y \in N(Y, R_\Delta)$. Then, there exists no $y' \in Y$ such that $y' + \Delta \leqq y$. Since $\Delta \leqq \Delta'$, there exists no $y' \in Y$ such that $y' + \Delta' \leqq y$ and $y \in N(Y, R_{\Delta'})$. □

From Definition 3, the preference relation R_Δ can be represented by a translated Pareto cone and is a generalization of several known approaches:

- $\Delta = (0, ..., 0)$. R_Δ is equivalent to the Pareto dominance and $N(Y, R_\Delta) = N(Y)$.
- $\Delta \leq (0, ..., 0)$. R_Δ is equivalent to an epsilon approximation [13].
- $\Delta \geq (0, ..., 0)$. R_Δ is equivalent to an approach presented in [2].

In these already known approaches, coordinates of Δ are either all positive, all negative or all zero. We present hereafter cases where Δ may contain positive, zero and negative components.

B. Interpretation

We illustrate a case where defining a constant threshold vector with positive and negative values is relevant with Example 1.

Example 1. Let $Y = \{y^1, y^2, y^3\}$ with $y^1 = (11, 9, 15)$, $y^2 = (10, 10, 18)$ and $y^3 = (15, 13, 12)$. The DM establishes that if a point y is significantly better than another point y' on the first two criteria and the difference on the third criterion is not too important, then y is preferred to y'. To model this situation, we set constant requirement values of 2 on criteria

f_1 and f_2 and a constant tolerance value of 5 on criterion f_3, resulting in a threshold vector $\Delta = (2, 2, -5)$.

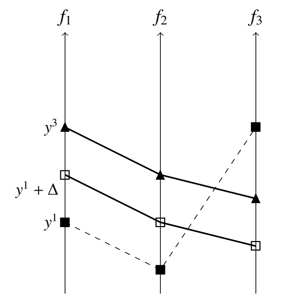

Fig. 1: y^1 R_Δ-dominates y^3 (Example 1).

Using this threshold, we have $y^1 R_\Delta y^3$ in accordance with the preference information provided by the DM (see Figure 1). As a consequence, y^3 does not belong to the R_Δ-nondominated set. Observe finally that we do not have $y^2 R_\Delta y^3$ because, in spite of better scores for y^2 on criteria f_1 and f_2, y^2 is largely worse than y^3 on criterion f_3. There are no other cases of R_Δ-dominance. Therefore, we have $N(Y, R_\Delta) = \{y^1, y^2\}$ whereas $N(Y) = \{y^1, y^2, y^3\}$.

The negative threshold implies that the relation R_Δ is not necessarily transitive. We illustrate such a case with Figure 2 in a biobjective space.

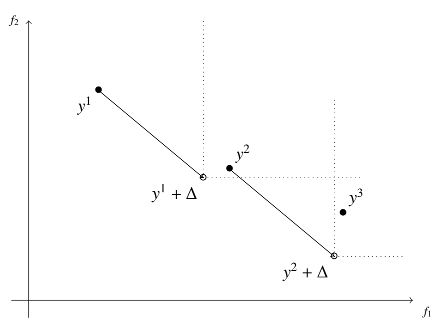

Fig. 2: A case where $y^1 R_\Delta y^2$ and $y^2 R_\Delta y^3$ but not $(y^1 R_\Delta y^3)$.

Non-transitive preferences can occur quite naturally in some cases. For instance, in Figure 2, y^1 is preferred to y^2 since the advantage of y^2 on criterion f_2 is judged negligible whereas the advantage of y^1 on criterion f_1 is judged significant. Similarly y^2 is preferred to y^3. It appears, however, that y^1 is not preferred to y^3 since the advantage of y^3 on criterion f_2 is no longer negligible.

When comparing two points, a deterioration on some criteria can be compensated by a relatively good improvement on other criteria. In Example 1, this deterioration is limited by a finite negative threshold. However it is possible not to constrain this deterioration. Indeed, the DM could determine that large improvements on a group of criteria are significant enough and the remaining criteria do not take part in the comparison. With Example 2, we illustrate that R_Δ can be useful to represent this notion of non-compensation.

Example 2. Let y^1, y^2, y^3 be the points defined in Example 1 and y^4 be a point of Y such that $y^4 = (20, 21, 6)$. The DM declares that a large improvement on criteria f_1 and f_2 cannot be compensated by any performance on criterion f_3. The threshold for the first two criteria is then greater than before and evaluated to 10. By defining $\Delta = (10, 10, -\infty)$, we get the following results:

$$y^2 + \Delta = \begin{pmatrix} 20 \\ 20 \\ -\infty \end{pmatrix} \leqq \begin{pmatrix} 20 \\ 21 \\ 6 \end{pmatrix}$$

Therefore we can conclude that y^2 R_Δ-dominates y^4 even if y^4 has a much better performance on the last criterion.

The R_Δ-nondominated set of a MOO problem can contain dominated points. We illustrate such a case in Example 3.

Example 3. Let y, y' be two points of Y such that $y = (11, 11, 17)$ and $y' = (10, 11, 9)$, and Δ be a constant threshold vector such that $\Delta = (2, 2, -6)$. We have $y' \leq y$ but not $(y' R_\Delta y)$ since

$$y' + \Delta = \begin{pmatrix} 12 \\ 13 \\ 3 \end{pmatrix} \nleqq \begin{pmatrix} 11 \\ 11 \\ 17 \end{pmatrix}$$

Thus y and y' belong to the R_Δ-nondominated set even if y is a dominated point.

In order to exclude Pareto dominated points from our set $N(Y, R_\Delta)$, we introduce another preference relation, which is the union of Pareto dominance and R_Δ .

C. Union of preference relations

We first introduce a result on the nondominated set of a union of preference relations.

Proposition 5. *Let R, R_j, $j = 0, 1, ..., m$, be binary relations, such that $R = \bigcup_{j=0}^{m} R_j$. Then, we have:*

$$N(Y, R) = \bigcap_{j=0}^{m} N(Y, R_j).$$

Proof. Let y be a point of $N(Y, R)$. Then, there is no point y' in Y, such that y is R_j-dominated by y', $j = 0, 1, ..., m$. Therefore y is in $N(Y, R_j)$, $j = 0, 1, ..., m$. The converse is also true. □

In addition to use the Pareto dominance, by setting $R_0 = \leq$, the DM could also be able to define more than one situation where he/she is able to establish a preference between two points. In a context of group decision making, several DMs are involved in the decision making process. Each DM could be able to propose a threshold vector so that all points of view are taken into account. Therefore we define a preference relation R_U as follows.

Definition 6. Let y, y' be two points of Y and Δ^i be threshold vectors in $\mathbb{R}^p$, $i = 1, ..., m$.

$yR_U y'$ if and only if $y \leq y'$ or $yR_{\Delta^i}y'$ for some $i = 1, ..., m$.

We ensure with the following corollary of Proposition 5 that the R_U-nondominated set does not contain dominated points.

Corollary 7. *Let Y be a set of points. Then, we have:*

$$N(Y, R_U) \subseteq N(Y).$$

Proof. The proof is straightforward from Proposition 5, where R_0 is the Pareto dominance. □

With Corollary 7, we also show that the R_U-nondominated set contains fewer points that the nondominated set and justifies the integration of preference information.

Corollary 8. *Let Y be a set of points in $\mathbb{R}^p$ and R_U be defined with m thresholds $\Delta^i, i = 1, ...m$, and Δ^{m+1} be a threshold vector in $\mathbb{R}^p$. Then, we have:*

$$N(Y, R_U \cup R_{\Delta^{m+1}}) \subseteq N(Y, R_U).$$

Proof. From Proposition 5. □

With Corollary 8, we show that the more you use preference information the less you have R_U-nondominated points. However there could be cases where the R_U-nondominated set is empty. We illustrate such a case in the following example.

Example 4. Let Y be a set of points in $\mathbb{R}^p$ containing two nondominated points y^1, y^2 such that $y^1 = (10, 10, 18)$ and $y^2 = (15, 13, 12)$. Let $\Delta^1 = (2, 2, -6)$ and $\Delta^2 = (-5, -4, 4)$ be two threshold vectors. Since $y^1 R_{\Delta^1} y^2$ and $y^2 R_{\Delta^2} y^1$, $N(Y, R_U)$ is empty.

As shown in Proposition 9, a sufficient condition for the existence of the R_U-nondominated set is:

Condition 1. There exists an objective $j \in \{1, ..., p\}$ such that for each threshold $\Delta^i, i = 1, ..., m$, $\Delta^i_j > 0$.

Condition 1 imposes the existence of at least one criterion that cannot be deteriorated in any preference situation. We introduce the following result on $N(Y, R_U)$ under Condition 1.

Proposition 9. *Let Y be a set of points in $\mathbb{R}^p$ and R_U be a preference relation defined by m thresholds. Under Condition 1, we have $N(Y, R_U) \neq \varnothing$.*

Proof. By Corollary 7, we know that $N(Y, R_U) \subseteq N(Y)$. We assume by contradiction that $N(Y, R_U) = \varnothing$. Therefore there exists $y^1, ..., y^q$ in $N(Y)$ such that $y^1 R_{\Delta^{i_1}} y^2 R_{\Delta^{i_2}} ... y^q R_{\Delta^{i_q}} y^1$, $i_1, ..., i_q \in \{1, ..., m\}$. Under Condition 1, there exists an objective $j = 1, ..., p$ such that $\Delta^i_j > 0, i = 1, ..., m$. Thus, $y^1_j + \Delta^{i_1}_j \leq y^2_j < y^2_j + \Delta^{i_2}_j \leq ... \leq y^q_j < y^q_j + \Delta^{i_q}_j \leq y^1_j$. This contradicts the existence of a circuit in relation R_U, hence the proposition. □

D. Elicitation framework for thresholds

The elicitation of a threshold vector Δ can be performed by identifying specific preference situations as in Example 1. In each of these situations, the DM can determine two groups of criteria, being the group of most and the group of least important criteria, respectively. The thresholds corresponding to the most important criteria are positive, while thresholds corresponding to the least important criteria are negative. This reflects the fact that one is more demanding on the most important criteria, whereas one is more tolerant on the least important criteria.

We propose a framework to establish different cases of dominance between two points. The DM is asked to evaluate different levels (e.g. small, average, large) of improvement and deterioration on each criterion. After determing these values (variable or constant), the definition of thresholds can be performed by listing typical situations, combining improvements and deteriorations, where a preference is clearly established. This approach easily handles heterogeneous criterion scales since the DM is asked to establish dominance based on strengths of improvement and deterioration. We give an example of the elicitation procedure with variable thresholds in the following.

Example 5. We introduce hereafter a situation with three objectives, where the DM defines two groups $M = \{1, 2\}$, and $L = \{3\}$, such that the DM considers objectives in M to be more important than the objective in L. Therefore the DM is interested in even moderate improvements for objectives in M, while only large improvements are considered for the objective in L. The values of improvements and deteriorations for each objective are illustrated in Figure 3.

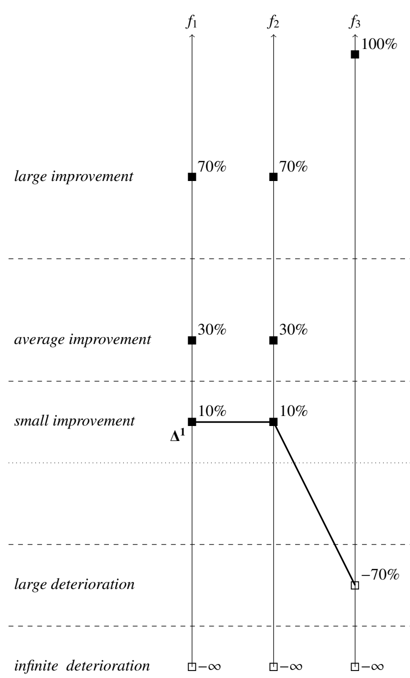

Fig. 3: Illustration of an improvement/deterioration scale for each criterion.

First, the DM determines that a point y is preferred to another point y' if the performances of y on the criteria in M reflect at least a small improvement compared to y' and provided that the difference on the objective in L is not greater than a strong deterioration. To model this, we can define the threshold vector $\Delta^1 = (10\%, 10\%, -70\%)$, which is represented by the thick line in Figure 3. Furthermore, the DM states that a point y is preferred to another point y' if the performances of y on the criteria in M reflect at least an average improvement compared to y' on both objectives in M, irrespective of the performances on the objective in L. Difference on the objective in L is thus not taken into account. Consequently, we introduce the threshold vector $\Delta^2 = (30\%, 30\%, -\infty)$.

However, the DM also prefers points that are good on at least two objectives. If a point y' is good on only one objective, namely 1 (respectively 2), there could exist a point y, whose performances on objectives $2, 3$ (respectively $1, 3$) are much better than for y'. This corresponds to large improvements of y' on objectives $2, 3$ (respectively $1, 3$). In these cases, y is preferred to y'. Large improvements on these pairs of objectives are modeled by the following threshold vectors $\Delta^3 = (70\%, -\infty, 100\%)$, $\Delta^4 = (-\infty, 70\%, 100\%)$. Observe that, since objectives $1, 2$ are more important than objective 3, an improvement on these objectives is more important than an improvement on the last objective. Therefore the value of large improvement on these objectives is smaller than the one on the last objective.

In the following, we present a framework to compute the R_U-nondominated set in discrete MOO algorithms. After reminding the general principles of these algorithms, we explain the additional steps to compute the R_U-nondominated set.

IV. Adaptation to discrete Multi-Objective optimization algorithms

A. *Presentation of discrete multi-objective optimization*

In discrete MOO, most solution approaches iteratively generate candidate solutions. A pool of solutions is updated when a new candidate solution arrives. The candidate solution is either inserted or discarded. It can also remove solutions already in the pool. Furthermore the images of solutions in the pool can guide the enumeration in the so-called *search region* [8], i.e. that part of the objective space that may still contain nondominated points. The main steps for the computation of the nondominated set are described in Algorithm 1 hereafter.

Algorithm 1 Generation algorithm

Require: Initialization of the search region.
  while The search region is not empty **do**
    Generate a nondominated point y in the search region.
    Update the search region by removing the part dominated by y.
  end while

A trivial approach to compute the R_U-nondominated set would be to use a classic MOO algorithm to generate the nondominated set and filter this set using pairwise comparisons in order to obtain the R_U-nondominated set (Corollary 7). However the computation time of the nondominated set increases with the size of the instance. Therefore we use the preference information to guide the search within the objective space, in order to have a gain in computation time. We adapt the main steps of Algorithm 1 to directly compute the R_U-nondominated set. Such methods iteratively find a nondominated point and then update the search region. We first present the adaptation of the updating step when a point is generated (Section IV-B). Then, we present the adaptation of the generating step of a R_U-nondominated point (Section IV-C).

B. *Updating the search region*

Most approaches designed to generate the nondominated set iteratively update a search region containing the remaining nondominated points. Given any generated point y in Y, they provide a procedure to remove from the search region the part that y dominates according to the Pareto dominance. The set of generated points is denoted by N and its associated search region by $S(N)$.

$$S(N) = \{y \in \mathbb{R}^p : \text{there is no } y' \text{ in } N \text{ such that } y' \leqq y\}$$

Moreover given a point y in Y, we denote by $D(y)$ the part of the objective space dominated by y.

$$D(y) = \{y' \in \mathbb{R}^p : y \leqq y'\}$$

Thus, updating the search region $S(N)$ with a point y can be reformulated as follows:

$$S(N \cup \{y\}) = S(N) \backslash D(y)$$

More generally we denote by $S_R(N)$ and $D_R(y)$ the search region associated to the preference relation R and the part of the objective space that y R-dominates, respectively.

$$S_R(N) = \{y \in \mathbb{R}^p : \text{there is no } y' \in N \text{ such that } y'Ry\}$$

$$D_R(y) = \{y' \in \mathbb{R}^p : yRy'\}$$

Thus, updating the search region $S_R(N)$ by a newly generated point y can be reformulated as follows:

$$S_R(N \cup \{y\}) = S_R(N) \backslash D_R(y)$$

We introduce a trivial result concerning the region R_Δ-dominated by a point y.

Proposition 10. *Let y and Δ be a point and a threshold vector in $\mathbb{R}^p$, respectively. Then, we have $D_{R_\Delta}(y) = D(y + \Delta)$.*

Proof. Trivial. □

The following corollaries are deduced from Proposition 10.

Corollary 11. *Let y and Δ be a point and a threshold vector in $\mathbb{R}^p$, respectively. Then, we have:*

$$D_{R_U}(y) = D(y) \cup D(y + \Delta)$$

Therefore we introduce the following formulation for the search region $S_{R_U}(N)$ associated to the preference relation R_U.

Corollary 12. *Let y and Δ be a point and a threshold vector in $\mathbb{R}^p$ respectively and N be a set of points in $\mathbb{R}^p$. Then,*
$$S_{R_U}(N \cup \{y\}) = S_{R_U}(N) \backslash (\ D(y) \cup D(y+\Delta)\ )$$

Each time a point y is generated, most of the recent generic discrete MOO algorithms (see, e.g., [7], [10]) include a procedure which takes y as an input and removes from the search region the part of the objective space that y dominates according to Pareto. From Corollary 12, updating the search region associated to R_U can be done with the same procedure by not only updating with the original point y but also updating with an artificial point $y + \Delta$. We illustrate the result above with Figure 4.

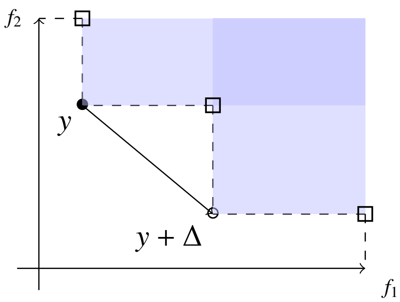

Fig. 4: After generating y, the part R_U-dominated by y is removed from the search region.

Extending this procedure to several thresholds $\Delta^1, ..., \Delta^m$ is straightforward. After generating a point y, the search region is updated with y and m artificial points $y + \Delta^1$, ..., $y + \Delta^m$. We now focus on the generation of a new R_U-nondominated point.

C. Generating a R_U-nondominated point

The search region can be decomposed as a list of *search zones* [8]. Each search zone is induced by a local upper bound u in $\mathbb{R}^p$ and is defined as follows.

$$\{y \in \mathbb{R}^p : y_i < u_i,\ i = 1, ..., p\}$$

The recent discrete MOO algorithms mentionned above maintain, more or less explicitly, a list of search zones that represents the search region. Generating a nondominated point is done by choosing a search zone in the list and generating a nondominated point in this search zone if it exists. In the following, we focus on the computation of a R_U-nondominated point in a search zone denoted by its local upper bound u in the objective space.

Exploring a search zone can be done by solving the following program with λ a vector of strictly positive weights in $\mathbb{R}^p$:

$$P^u \begin{cases} \min & \sum_{i=1}^{p} \lambda_i y_i \\ s.t. & y \quad \in \quad Y \\ & y_i \quad < \quad u_i, \quad i = 1, ..., p \end{cases}$$

Remark 1. Problem P^u involves strict inequalities $y_i < u_i, i = 1, ..., p$, that must be transformed into large inequalities of the type $y_i \leq u_i - \varepsilon$ where ε is a small enough value. In our experiments, the objectives take integer values and we set $\varepsilon = 1$.

Two cases can occur:

- Program P^u is infeasible, in which case the search zone corresponding to u is retrieved from the search region;
- Program P^u yields a nondominated point y^*. However y^* can be R_{Δ^i}-dominated, $i = 1, ..., m$, by a point outside the search zone u as illustrated in Figure 5.

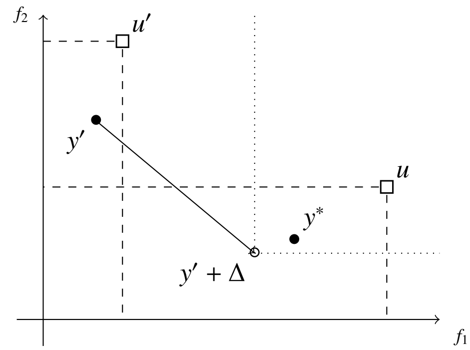

Fig. 5: Point y^* is R_U-dominated by y', which does not belong to the zone induced by u.

For the sake of clarity, we first present the generation of a R_U-nondominated point with one threshold vector Δ and then explain the extension to several threshold vectors Δ^i, $i = 1, ..., m$.

1) Case with one threshold:

Proposition 13. *Let y^* be the optimal point of P^u, λ be a vector of strictly positive weights in $\mathbb{R}^p$ and P^*_Δ be the following program:*

$$P^*_\Delta \begin{cases} \min & \sum_{i=1}^{p} \lambda_i y_i \\ s.t. & y \quad \in \quad Y \\ & y_i + \Delta_i \quad \leq \quad y^*_i, \quad i = 1, ..., p \end{cases}$$

1) *If P^*_Δ is infeasible, then $y^* \in N(Y, R_U)$;*
2) *If P^*_Δ is feasible, then $y^* \notin N(Y, R_U)$.*

Proof.

1) If P^*_Δ is infeasible, there exists no point y' in Y such that $y' + \Delta \leqq y^*$. Therefore y^* is R_U-nondominated.
2) If P^*_Δ is feasible and y' is its optimal point, $y' + \Delta \leqq y^*$. Therefore y^* is R_Δ-dominated by y'.

□

Proposition 13 provides two rules when generating a nondominated point y^*.

- If y^* is R_Δ-nondominated, then y^* is added to N and the search region $S_{R_U}(N)$ is updated with y^* and $y^*+\Delta$ (Section IV-B).
- If y^* is R_Δ-dominated by a point y', then y^* is not added to N but the search region $S_{R_U}(N)$ is updated with $y'+\Delta$ to avoid enumerating y^* again, and $y^*+\Delta$ since there could be points that are R_Δ-dominated by y^* but not by y' (see Figure 6).

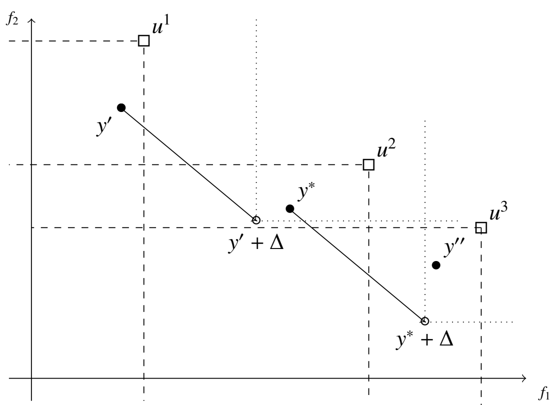

Fig. 6: The point y'' is R_U-dominated by y^*, even if y^* is also R_U-dominated.

Remark 2. Note that using y^* to update the search region is redundant with using $y'+\Delta$. Observe also that y' can still be generated in further iterations since $y'+\Delta$ does not prune y'.

Remark 3. The linear formulation of Δ (Definition 2) is helpful, since it only adds linear constraint to the problem.

2) Case with several thresholds:

We extend the generating step for several threshold vector $\Delta^i, i=1,...,m$. We check for each threshold $\Delta^i, i=1,...,m$, if the point is R_{Δ^i}-dominated by solving the problem $P^*_{\Delta^i}$ described in Proposition 13. There are two possible situations:

- If for all $i=1,...,m$, there exists no feasible point for all $P^*_{\Delta^i}$, then y^* is R_U-nondominated. Therefore we add y^* to the R_U-nondominated set. The search region is updated with $y^*, y^*+\Delta^1,...,y^*+\Delta^m$.
- If, for some $k=1,...,m$ the point y^* is R_{Δ^k}-dominated by a point y', y^* is not added to the R_U-nondominated set . Programs $P^*_{\Delta^{k+1}},...,P^*_{\Delta^m}$ are not solved since y^* is already proven to be R_U-dominated. The search region is updated with $y'+\Delta^1,...,y'+\Delta^m$ and also $y^*+\Delta^1,...,y^*+\Delta^m$ for the same reason as in the specific case with one threshold vector (see Figure 6).

Observe that both updating the search region and computing a R_U-nondominated point require a larger computational burden than for the Pareto nondominated set. Indeed with m threshold vectors, updating the search region requires m additional updates and 1 to m additional optimizations. However the preference information conveyed by these thresholds usually leads to a R_U-nondominated set which is substantially smaller than the Pareto nondominated set. This is illustrated in Section V.

V. Numerical experiments

We performed our experiments using the generic algorithm presented in [11], which refines Algorithm 1 and maintains a list of search zones to describe the search region. This algorithm generates all nondominated points for discrete MOO problems. We have generated our results on the Multi-Objective Assignment Problem (MOAP). The MOAP consists in assigning n agents to n tasks in order to minimize the total assignment cost. An agent is assigned to one and only one task and a task is assigned to one and only one agent. Each agent-task assignment involves p costs. The total cost of an assignment is computed by adding up the costs of every chosen agent-task assignment. These objective costs are random integers uniformly drawn in the interval $[1,20]$.

For MOAP we used the following instance sizes:

- $p=3$ and $n=30$ (3-MOAP30);
- $p=3$ and $n=40$ (3-MOAP40);
- $p=3$ and $n=50$ (3-MOAP50).

The thresholds used for the experiments are the one used in Example 1.

- Δ^1 with $a^1=(10\%,10\%,-70\%)$ and $b^1=(0,0,0)$;
- Δ^2 with $a^2=(30\%,30\%,0)$ and $b^2=(0,0,-\infty)$;
- Δ^3 with $a^3=(70\%,0,100\%)$ and $b^3=(0,-\infty,0)$;
- Δ^4 with $a^4=(0,70\%,100\%)$ and $b^4=(-\infty,0,0)$.

We computed the Pareto nondominated set and the R_U-nondominated set with R_U being the union of the Pareto dominance and $R_{\Delta^1}, R_{\Delta^2}, R_{\Delta^3}, R_{\Delta^4}$.

A computer with a Linux Debian operating system, 3.2 GHz processor and a 16 GB memory limit has been used for the experiments. CPLEX 12.6.3 is the solver provided for the algorithm. We used 10 instances for each problem size to compute our results. We report the average CPU time (s), the average size of the different R-nondominated set, denoted by $|N(Y,R)|$, R being either Pareto dominance or R_U. Since R_U does not satisfy Condition 1, we also give the number of empty R_U-nondominated sets (# Empty) in Table I, II, III.

TABLE I: CPU Time and Number of Points for 3-MOAP30 Instances.

3-MOAP30	**CPU time(s)**	$\mathbf{\|N(Y,R)\|}$	# Empty
Pareto	1424.61	6181.4	-
R_U	325.53	443.9	0

TABLE II: CPU Time and Number of Points for 3-MOAP40 Instances.

3-MOAP40	**CPU time(s)**	$\mathbf{\|N(Y,R)\|}$	# Empty
Pareto	4774.81	14679.7	-
R_U	800.71	562.6	0

TABLE III: CPU Time and Number of Points for 3-MOAP50 Instances.

3-MOAP50	CPU time(s)	$\lvert\mathbf{N(Y,R)}\rvert$	# Empty
Pareto	11753.18	24916.8	-
R_U	1602.90	553.8	0

We observe that for all instances the gain in computation time is significant. The number of R_U-nondominated points is also reduced compared to the Pareto nondominated set. The ratio concerning the difference of size of each nondominated set does not correspond to the ratio on the difference of computation time. Indeed, there are between 1 and 4 additional optimizations to check for R_U-dominance. Therefore the time spent for each generated R_U-nondominated point is larger than the time spent for a Pareto nondominated point. Observe that even if $\Delta^1, \Delta^2, \Delta^3$ and Δ^4 do not satistfy Condition 1, there is no empty set in all instances.

In order to evaluate the quality of the returned R_U-nondominated set, and its consistency with the threshold vectors, we compare it with the Pareto nondominated set. For this purpose, after computing these two sets, we take as indicators the minimum and maximum values on each objective over each of these two sets. These values correspond to the ideal and nadir point values, respectively. We report the results on one 3-MOAP40 instance in Table IV.

TABLE IV: Ideal and Nadir Points for a 3-MOAP40 Instance.

3-MOAP40 instance	Ideal			Nadir		
	f_1	f_2	f_3	f_1	f_2	f_3
Pareto	49	52	55	507	509	489
R_U	101	92	309	188	171	466

Several observations can be made on this instance. The threshold vectors $\Delta^1, \Delta^2, \Delta^3$ and Δ^4 represent a situation where f_1 and f_2 are more important than f_3. Indeed, the maximum value on f_1 and f_2 have significantly decreased, while the maximum performance on f_3 has only slightly decreased. On the contrary, the ideal point has only slightly increased on f_1 and f_2 in comparison with f_3. This underlines the fact that good performances on f_3 are less considered. Observe that the deterioration on f_1 and f_2 is the consequence of avoiding points being good on only one objective. Finally, note that the modifications of the ideal and nadir point on f_1 and f_2 are comparable, which seems natural since objectives f_1 and f_2 are both considered of similar importance.

VI. Conclusions

We presented an original preference relation based on requirement and tolerance values, that translate the Pareto dominance. This not necessarily transitive preference relation can have a variable preference structure and integrates the notion of non-compensation. To use this preference relation, we based our approach on a general scheme of multi-objective discrete optimization algorithms. After testing on several instances of MOCO problems, we observe that the results are promising.

This work offers several research directions such as eliciting methods for the thresholds or use a set of thresholds defined by constraints. Applying this preference relation with approximation algorithms such as, e.g., multi-objective evolutionary algorithms could be useful in the case where the computational effort is too demanding for exact algorithms. Covering sets [12] could also be studied in this framework when the nondominated set according to the preference relation is empty.

References

[1] B. S. Ahn, 2007. *The OWA aggregation with uncertain descriptions on weights and input arguments.*, IEE Transaction on fuzzy systems, 15(6):1130–1134.

[2] A. Engau and M. M. Wiecek, 2007. *Exact generation of epsilon-efficient solutions in multiple objective programming*, OR Spectrum, 29(2):335–350.

[3] Fishburn, P. C., 1965. *Analysis of decisions with incomplete knowledge of probabilities*. Operations Research 13 (2):217–237.

[4] L. Gardiner and D. Vanderpooten, 1997. *Interactive multiple criteria procedures: Some reflections*. In: Clímaco, J. (Ed.), Multicriteria Analysis. Springer Verlag, Berlin, pp. 290–30

[5] M. Grabisch, 1996. *The application of fuzzy integrals in multicriteria decision making*, European Journal of Operational Research, 89(3):445–456.

[6] B. Hunt, M. M. Wiecek and C. S. Hughes, 2010. *Relative importance of criteria in multiobjective programming: A cone-based approach*. European Journal of Operational Research, 207 (2):936–945.

[7] G. Kirlik and S. Sayin, 2014. *A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems*. European Journal of Operational Research, 232 (3):479–488.

[8] K. Klamroth, R. Lacour and D. Vanderpooten, 2015. *On the representation of the search region in multi-objective optimization*. European Journal of Operational Research, 245 (3):767–778.

[9] K. Miettinen and MM.. Mäkelä, 2000. *Interactive multiobjective oprimization system WWW-NIMBUS on the Internet*. Computers & Operations Research, 27 (7):709–723.

[10] J. Sylva and A. Crema, 2008. *Enumerating the set of non-dominated vec- tors in multiple objective integer linear programming*. RAIRO - Operations Research, 42 (3):371387.

[11] S. Tamby, 2017. *Approches génériques pour la programmation linéaire en nombres entiers multi-objectif*, Department of Computer Science, Paris-Dauphine University.

[12] D. Vanderpooten, L. Weerasena and M.M. Wiecek, 2016. *Covers and approximation in multiobjective optimization*. Journal of Global Optimization. doi: 10.1007/s10898-016-0426-4.

[13] D.J. White, 1986. *Epsilon efficiency*. Journal of Optimization Theory and Applications, 49 (2):319–337.

[14] A.P. Wierzbicki, 1986. *On the completeness and constructiveness of parametric characterizations to vector optimization problems*. OR-Spektrum, 8 (2):73–87.

[15] R. R. Yager, 1988. *On ordered weighted averaging aggregation operators in multicriteria decisionmaking*. IEEE Transactions on Systems, Man and Cybernetics, 18 (1):183–190.