



**HAL**  
open science

## A taxonomy of clustering procedures

Olivier Cailloux, Claude Lamboray, Philippe Nemery

► **To cite this version:**

Olivier Cailloux, Claude Lamboray, Philippe Nemery. A taxonomy of clustering procedures. 66th Meeting of the European Working Group on MCDA, 2007, Marrakech, Morocco. pp.N/A. hal-00985860

**HAL Id: hal-00985860**

**<https://hal.science/hal-00985860>**

Submitted on 30 Apr 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A taxonomy of clustering procedures

CAILLOUX, Olivier\*; LAMBORAY, Claude & NEMERY, Philippe

Université Libre de Bruxelles  
Faculté des Sciences Appliquées - Ecole Polytechnique  
Service de Mathématiques de la Gestion (SMG)  
CoDE : <http://code.ulb.ac.be>  
Boulevard du Triomphe CP 210/01  
1050 Brussels  
Belgium

August 21, 2007

## Abstract

In the field of multicriteria decision aid, considerable attention has been paid to supervised classification problems, especially to so-called sorting problems, where an order is assumed on the predefined classes. Recently, some non-supervised multicriteria classification procedures, also known as multicriteria clustering procedures, have been proposed aiming to discover data structures from a multicriteria perspective. We enlighten some properties of such approaches and their differences with regards to classical procedures, and we propose a taxonomy of this family of procedures. Moreover, we analyze extend to which these procedures differ from the multicriteria ranking problematic.

**Keywords:** Multicriteria clustering; Taxonomy; MCDA; Clustering; Multiple criteria analysis.

## 1 Introduction

Classification problems are commonly encountered in various application fields such as health care, biology, finance, marketing, agriculture, etc. [DHS01, Zop02]. Two main problems are usually distinguished: the supervised and the unsupervised classification or clustering problem.

In supervised classification problems the purpose is to assign a new object to homogeneous groups (so-called classes) which are defined a priori. We may think for instance about the medical diagnosis problem where a new patient has to be assigned to a known pathology-class on the basis of a set of symptoms. A lot of

---

\*corresponding author: [olivier.cailloux@ulb.ac.be](mailto:olivier.cailloux@ulb.ac.be)

supervised classification procedures have been proposed such as the k-nearest neighbor algorithm or the Bayes classifier [ELL01, DBF05].

On the other hand, there may be no information available about the groups (which are then often called clusters) and the purpose is then to elicit a structure in the data set. For instance, one may consider a marketing problem where the aim is to discover similar customer behaviors in the retail industry. Among the most common clustering procedures, one may cite the K-means, hierarchical or finite mixture densities algorithms [JMF99]. Multicriteria tools have also been considered. Bisdorff proposes an Electre-like clustering procedure based on the L-valued kernels [Bis02].

In the multicriteria decision aid field, a lot of attention has been paid to supervised classification. Moreover, a distinction is made between nominal classification and ordinal classification, also called sorting [ZD02b]. In the former case, there is no order on the classes whereas in the latter, the classes (also called categories) are completely ordered from the best to the worst. For instance, the evaluation of the creditworthiness of a company is a sorting problem. A bank will assign a request for a new credit into one of the following classes: the classes of accepted credits (the “creditworthy” clients), the classes of clients for which some more information is needed and the classes of refused credits (“untrustworthy” clients). In sorting problems the procedures explicitly take into account preference information (using criteria, weights, indifference and preference thresholds, etc.). Among others, we may cite Electre-Tri [Yu92], FlowSort [NL07], UTA-DIS [ZD02a].

Recently, some new multicriteria clustering procedures have been proposed who aim to discover data structures from a multicriteria perspective. De Smet and Guzman [SG04] and Figueira et al. [FSMB04] have extended the classical K-means in a multicriteria framework, explicitly taking into account a preference structure. The question of the positioning of these new research problems with respect to both the classical clustering field and multicriteria problematics such as ranking has, to the best of our knowledge, not been addressed yet. A first answer is provided in this paper.

To this aim, we define some properties of clustering procedures. The fact of working with criteria instead of attributes (like in classical approaches) is a first distinctive feature that is analyzed. Furthermore, obtaining a partition with a (possibly transitive) relation on the clusters is considered as well. Based on these considerations, we propose a taxonomy of clustering procedures.

Such a clarification opens up some perspectives for designing new multicriteria clustering procedures. We propose some broad strategies to achieve this. Besides, we discuss how such approaches could help a decision maker.

The paper is organized as follows. After having described the general context in which we place in this paper, we describe in section 3 our taxonomy proposition. We first propose a property to distinguish classical clustering from multicriteria clustering, then, propose to distinguish the case where the multicriteria clustering procedure contains a relation, then point the fact that this relation could

be transitive, and discuss the particularities of each of these branches. We compare in section 4 the multicriteria clustering to the ranking problematic, and we conclude by suggesting some paths which exploration may prove interesting.

## 2 Context

To simplify the context, we will restrict here to the clustering procedures taking attribute values in the set of reals numbers, and we restrict to crisp clustering, i.e. to the procedures assigning an object to exactly one cluster. Let us define the context more precisely.

**Definition 1** A **real-valued attribute function**  $g$ , defined in the context of a clustering procedure using a set of actions  $A = \{a_1, \dots, a_n\}$ , is a function taking its input in  $A$  and giving an output in  $\mathbb{R}$ .  $\square$

We call here “action” what is usually called in the classical clustering domain “object”, because the main concern of this paper is the multicriteria domain.

**Definition 2** A **real-valued clustering procedure** is a procedure which takes as input a set  $A$  of actions to be clusterized and a set  $F$  of real-valued attribute functions, and gives as output a partition  $\rho$  on the set of actions, or such a partition  $\rho$  and a binary relation on  $\rho$ .  $\square$

We use the symbol  $F$  to refer to what is usually called in the multicriteria decision aid field a criteria *family*. We include in this rather general definition the case where the procedure’s output includes a binary relation on the resulting partition. This is needed to include the Relational Multicriteria Clustering procedures, which are defined in Section 3.2.

We will study in this paper the procedures which are deterministic and can be modeled as a function  $f(A, F)$  giving the resultant partition  $\rho$  (and the resultant partition relation if applicable).

Some clustering procedures are not deterministic because they use heuristics to guess the possibly best partition (e.g., K-means). In that case, we will consider a deterministic version of the same procedure which always selects the best partition instead of the one chosen by the heuristic. We thus assume that some algorithm to deterministically break the ties is used. An example of such reasoning applied to a multicriteria clustering procedure is shown in Appendix A.

## 3 Taxonomy

### 3.1 Criteria dependency

Some clustering procedures [SG04, FSMB04] have been proposed in the multi-criteria decision aid domain. These procedures are a bit particular in the sense that they use criteria instead of attributes. To the best of our knowledge, no formal separation between these procedures and classical clustering procedures (using attributes without preferential information) has been proposed yet. In this section, we propose a way to differentiate these particular procedures from the classical ones, by the way of a property called “criteria-dependancy”.

A criteria is an attribute including a preferential information on the set of considered actions. For example, a criterion  $g$  is often defined using a binary transitive relation  $S_g$  on the set of actions such that for any two actions  $a$  and  $b$ , we have  $aS_gb$  if and only if  $a$  is preferred or indifferent to  $b$ .

In order to include situations where only a binary relation  $S_g$  is provided to describe a criterion, we assume here that such an ordinal criterion is first numerically represented: if  $S_g$  is a semi order, then a real-valued function  $g$  and a constant  $\epsilon$  can be defined such that  $g(a) \geq g(b) - \epsilon$  iff  $aS_gb$ . Such a transformation is always possible [PV97]. The function  $g$  can then be considered as an input of any real-valued clustering procedure.

It should be noted that a “direction” of preference on the set of actions can be deduced from the preference information associated with a criterion. For example, a “price” attribute, used for a seller, would become a “sell price” criterion, having the same values as the price, but including the information that it should be maximized. In a multicriteria procedure used by a buyer, the “buying price” (or cost) criterion should rather be used, where it should be minimized.

Formally, such a “criteria-dependant” procedure satisfies the definition given here above and is thus a real-valued clustering procedure. But not all real-valued clustering procedures are criteria-dependant procedures: to be so, a procedure must somehow take into account the criterion-nature of the attributes it is dealing with. This is reflected by the “criteria-dependancy” property that we define hereunder. This property captures the fact that the qualified procedure depends, or not, on the criteria it uses. It is satisfied when the procedure’s results sometimes change when the criteria information change.

Let us define the  $INV(F, g^*)$  function, which for any attribute set  $F$  and attribute  $g^* \in F$ , transforms  $F$  to another set of attributes, similar to the first, but where the attribute  $g^*$  has been inverted by inverting the sign of its attribute values.

$$\forall F, \forall g^* \in F : INV(F, g^*) = (F \setminus \{g^*\} \cup \{g^{*'}\}), \text{ where } g^{*'}(a) = -g^*(a) \forall a \in A.$$

In the case where the criterion which we invert has been initially ordinal (only a

binary relation  $S_{g^*}$  is provided), then this construction consistently inverts the order. Assume that  $aS_{g^*}b$  and therefore build a real-valued function  $g^*$  such that  $g^*(a) \geq g^*(b) - \epsilon$ . Applying the transformation to  $g^*$  yields a new criterion function  $g^{*'}$  such that  $g^{*'}(a) = -g^*(a)$  and  $g^{*'}(b) = -g^*(a)$ . That is why we now have that  $g^{*'}(b) \geq g^{*'}(a) - \epsilon$ . The criterion  $g^{*'}$  then models a semi-order  $S_{g^{*'}}$  such that  $bS_{g^{*'}}a$ .

If a clustering procedure does not use the added informations of the criteria, transforming the attributes set using the *INV* function will not change the procedure's output. In the opposite case, the procedure's output should change, at least for some input sets, because the contrary would mean that the preferential information is never really taken into account. It should be noted that the fact that the output changes does not tell us how the criteria information has actually been exploited. An example of a procedure whose output changes when the *INV* function is applied is given in Appendix B (see Figures 3 and 4).

**Definition 3** A real-valued clustering function  $f$  is **criteria-dependant** iff

$$\exists A, F, g^* \mid f(A, F) \neq f(A, INV(F, g^*)).$$

□

**Definition 4** A **Multicriteria Clustering procedure**, or MCC<sup>1</sup> procedure, is a real-valued clustering procedure which is criteria-dependant. Otherwise, the procedure is said to be a **Classical Clustering procedure**. □

Having a formal way to distinguish between these two types of procedures has an additional benefit: we can ensure whether a procedure used in the multicriteria domain, such as those ones already published in the literature, is really using the criteria information.

Indeed, it is interesting to note that some, if not most, Classical Clustering procedures define their partition quality measure by aggregating distances (which are symmetric measures) on each attribute, for example the least-square error quality measure, whereas well-known multicriteria procedures such as PROMETHEE [BVM86] (on which PROMETHEE Cluster is based) use a non-symmetric  $\pi$  preference function to discriminate between actions.

It is thus more natural for a Multicriteria Clustering procedure to be criteria-dependant than for a classical one. This reasoning is confirmed if looking at the MC-K-means procedure, which is proved to be criteria-dependant in Appendix B. But the link between being called multicriteria, or using non-symmetric preference functions such as  $\pi$ , and being criteria-dependant, is not guaranteed, as can be seen by examining the PROMETHEE-Cluster procedure. We prove in Appendix A that this procedure, although based on PROMETHEE, is in fact not criteria-dependant. Another example of a clearly non criteria-dependent procedure is presented by Bisdorff [Bis02] who, although using multicriteria tools, makes no use of preference directions. Instead, the procedure is based on a symmetric similarity index.

---

<sup>1</sup>We use in this paper the conventional "MC" abbreviation for the word "multicriteria".

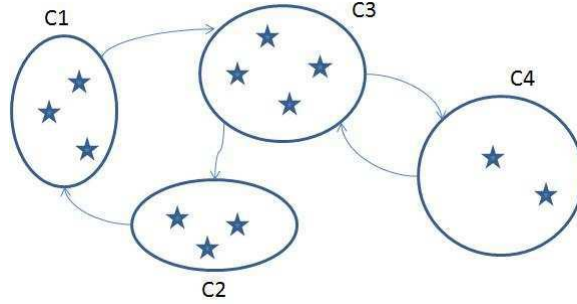


Figure 1: Example of a partition with a relation on the clusters.

### 3.2 Relational Multicriteria Clustering

Among all the criteria-dependent clustering procedures, the second point of interest is the presence of a relation between the clusters.

In the multicriteria context, actions are generally compared by defining a preference structure. A decision maker who compares two actions may express a preference, an indifference or an incomparability statement between these two actions [RV85]. It is thus natural to somehow incorporate this information into the clustering results.

Classical clustering procedures typically do not propose a preferential relation between the obtained clusters because they are not criteria-dependant. Indeed, the absence of preference statements between action pairs makes it difficult to infer a preference statement between two group of actions, i.e. clusters.

On the contrary, one may imagine to enrich a partition obtained by a MCC procedure with a binary preference relation between its clusters.

**Definition 5** A **Relational Multicriteria Clustering (RMCC) procedure** is a MCC procedure which outputs a partition  $\rho$  and a binary relation on  $\rho$ .  $\square$

This relation may for instance be an “at least as good” relation and will be noted  $S$ . In a general framework, this permits to obtain preferred, incomparable or indifferent clusters:  $C_1PC_2$  means that cluster  $C_1$  is preferred to cluster  $C_2$  ( $C_1SC_2$  and  $C_2\neg SC_1$ ),  $C_1IC_2$  that cluster  $C_1$  is indifferent to  $C_2$  ( $C_1SC_2$  and  $C_2SC_1$ ) and  $C_1JC_2$  that  $C_1$  and  $C_2$  are incomparable ( $C_1\neg SC_2$  and  $C_2\neg SC_1$ ).

In the example (see Fig. 1), we have that cluster  $C_1$  is preferred to cluster  $C_3$ ,  $C_3$  preferred to  $C_2$  and  $C_2$  to  $C_1$ . Finally we have that clusters  $C_3$  and  $C_4$  are indifferent.

Although the extension of the K-means proposed by De Smet and Guzman [SG04] (defined in Appendix B) explicitly uses a preference relation between the actions, there does not exist a relation on the clusters. The use of rich

preference information as input for achieving a classical “output” induces a loss of information and may be criticized. The two applications given in their paper illustrate this concern. For instance, in the first application, countries are grouped according to their level of creditworthiness. The resulting clusters are then treated as if they were ordered whereas no ordering appears in the results of the procedure.

The following strategy can be pursued in order to obtain a RMCC procedure. First, a classical clustering algorithm (such as the K-means algorithm) is applied on the data such that each cluster can be characterized with a centroid. In a second step, any multicriteria pairwise comparison procedure can be applied on these centroids in order to come up with an “at least as good” relation on the clusters. The outranking relation computed in ELECTRE I [RB93] can be used for this purpose.

### 3.3 Ordered Multicriteria Clustering

We treat in this section the special case when RMCC procedures always output a transitive relation on the clusters.

**Definition 6** An **Ordered Multicriteria Clustering (OMCC) procedure** is a RMCC procedure which outputs a partition  $\rho$  and a transitive binary relation on  $\rho$ .  $\square$

The transitivity property is special in the sense that it unambiguously implies an order on the clusters. If the relation is complete, then the clusters can be totally ranked. If the relation is not complete, then the transitivity property nevertheless implies a partial ranking, with some incomparable clusters.

OMCC procedures have the advantage over RMCC ones of providing less ambiguous results. However, the decision maker may feel uncomfortable with forcing transitivity. In such case, RMCC procedures would be more suitable. In a multicriteria decision aid context, compare this to either only computing pairwise comparisons (which may lead to a non-transitive relation) or going on and deriving a ranking from these pairwise comparisons.

Ordering clusters can be useful when some hierarchy has to be discovered in the data. Consider for instance the problem where employees, evaluated on various performance measures, have to be clustered. Depending on the data, one may discover for instance three clusters: a cluster with the above average performing employees, a cluster with the average performing employees and a cluster with the below average performing employees.

The procedure proposed by Nemery [NS05] is an example of what we call OMCC. It combines the ideas of both the clustering and the ranking problematics. First pairwise preference degrees are computed. For any ordered partition, a homogeneity indicator measures to what extent the actions belonging to the same cluster are similar and a heterogeneity indicator measures to what extent the order on the clusters is compatible with the preference degrees.



Apart from this specific procedure, the strategy described in Section 3.2 can also be adapted in order to build OMCC procedures: first, cluster the data, then, rank the centroides using a multicriteria ranking procedure. Conversely, a ranking procedure can be first applied on the data, then an ordered partition compatible with that ranking can be built, effectively splitting up the initial ranking.

OMCC procedures are also useful as a preprocessing step before applying a sorting procedure, such as for instance Electre-Tri [Yu92]. In fact, the following OMCC procedure could be imagined which is directly based on the sorting procedure. Suppose that all the parameters of that sorting procedure, except the parameters defining the classes (for example the reference profiles in the case of Electre-Tri) have been fixed by the decision maker. After generating a set of reference profiles, the sorting procedure can be applied and each action can be assigned to a class. A quality measure needs to be defined which evaluates how “easy” these reference profiles allow to assign the actions. The resulting OMCC procedure is the one which chooses the optimal reference profiles set, i.e. the one which yields the best assignments after applying the sorting procedure.

Apart from an ordered partition, such a procedure also provides good reference profiles that come with the optimal ordered partition. This could support a decision maker who would like to use a sorting procedure but is uncomfortable with the definition of the classes.

## 4 Comparison with the ranking problematic

Ordered clustering is closely related to the problematic of ranking. According to B. Roy [Roy85], the ranking problematic consists in partitioning the set of actions into partially or totally ordered equivalence classes. By adopting such a rather general definition, OMCC can be considered as a particular case of the ranking problematic. In fact, an OMCC procedure, which partitions the actions into ordered classes, can very naturally be seen as just another ranking procedure. Despite this apparent similarity between OMCC and ranking, let us however insist on some fundamental differences between these two problems.

On the one hand, a ranking procedure aims at discriminating the different actions. Consequently, a reasonable ranking procedure tends to maximize the number of equivalence classes: the preferred case in a ranking procedure is usually to build a linear order whenever possible. In such a case, the number of classes is maximal since it corresponds to the number of actions. Even if the number of equivalence classes is not always equal to the number of actions, the size of an equivalence class tends to be as small as the input data allows it to be.

On the other hand, a clustering procedure also aims at discriminating the different actions but at the same time it also tries to group together similar actions. Whereas the first objective tends to maximize the number of classes, the second objective tries to minimize it. The clustering solution finally adopted can usu-

Reference	Type
[Bis02]	CC
[FSMB04]	CC
[SG04]	NRMCC
[NS05]	OMCC

Table 1: Some clustering procedures.

ally be seen as a compromise between these two contradictory objectives. This observation will have several consequences.

Let us imagine a trivial problem with one criterion. Applying any reasonable ranking procedure to this data, we expect the output to be exactly the preference relation given by that criterion. Indeed, it is hardly conceivable that two actions are in the same equivalence class in the output ranking when the input data clearly states that the first action is preferred to the second one. Applying a OMCC procedure to the trivial one criterion problem, one could however imagine that the output is different from the initial criterion since some actions can possibly be put together. For example in a problem with three actions, where  $g(a) = 1, g(b) = 2$  and  $g(c) = 1000$ ,  $c$  being preferred by far to  $b$  and  $b$  being slightly preferred to  $a$ ,  $a$  and  $b$  could be put together.

This example shows another difference between a solution obtained with a ranking procedure and one obtained with an OMCC procedure. In the first case, using the semantics of a  $\{P, I, J\}$  preference structure [RV85], an equivalence class can be interpreted as a set of actions which are indifferent between themselves. Consequently, there is no preference nor incomparability between actions of the same equivalence class. In the second case, there is no clear link anymore between the cluster obtained in the OMCC procedure and such an indifference relation. In the example above,  $a$  and  $b$  could belong to the same cluster, even so  $a$  is preferred to  $b$ .

Moreover, in a ranking procedure, the decision maker usually does not fix the number of equivalence classes: it will be implicitly determined by that particular ranking mechanism. It does not usually make sense to fix it because it should be maximized. This is not true in the OMCC domain, where some procedures require to fix the number of clusters in advance.

## 5 Conclusion

The taxonomy of clustering procedures which we have presented in this paper is summarized in Figure 2. First criteria-dependency is used to distinguish between Classical Clustering and what we call Multicriteria Clustering. We then further classify MCC procedures into those which do not additionally yield a preference structure on the clusters and those which do. This distinction leads to either Non-Relational Clustering or Relational Clustering. Finally, the transitivity of this possible preference structure on the clusters characterizes Ordered

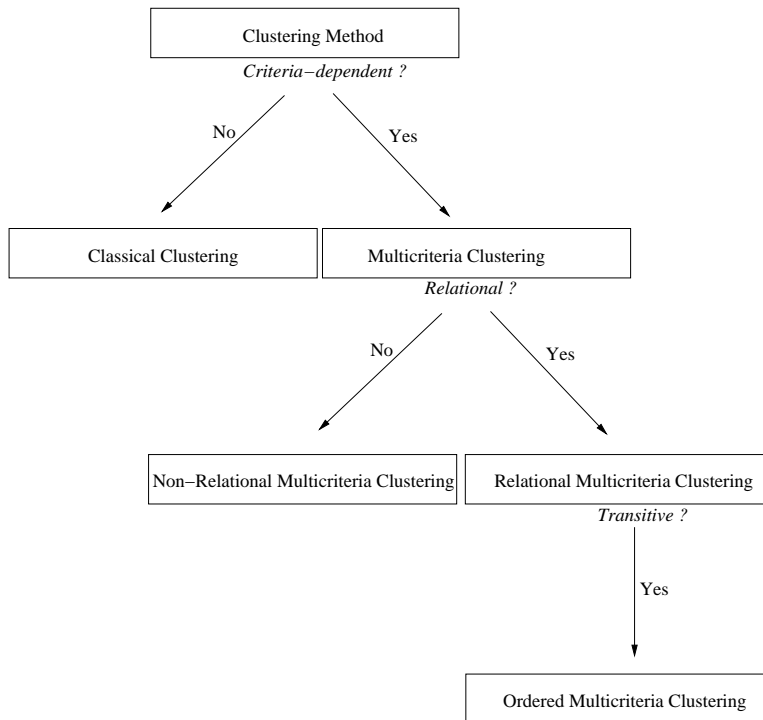


Figure 2: A taxonomy of clustering procedures.

Multicriteria Clustering. We recapitulate in Table 1 how some clustering procedures developed by the multicriteria community fit into this taxonomy.

The main goal of this paper was to clarify the meaning and the distinctive features of Multicriteria Clustering. Criteria-dependency is an important aspect since it imposes that the clustering procedure takes into account the preference scale implied by the criteria. Surprisingly, a procedure such as PROMETHEE Cluster, which nevertheless assumes a criteria input, is not criteria-dependent.

Non-relational Multicriteria Clustering procedures are peculiar since, on the one hand, they take into account the preferential information contained in the criteria, but, on the other hand, they output a result without any preferential information between the clusters. Seen from this angle, the interpretation of the partition obtained by these type of procedures is not entirely clear yet.

Relational Multicriteria Clustering procedures overcome this limitation by providing a preference structure on the clusters. However, only a few RMCC procedures have been proposed so far. The next step will now be to actually develop particular RMCC procedures (OMCC or not), following for instance one of the strategies explained in this paper. This will also help to analyze and understand more easily the results obtained by this type of clustering procedures. For instance, using OMCC procedures raises the question of monotonicity. The impact of improving some evaluations on the resulting ordered partition could be investigated.

Finally, the different types of clustering procedures defined in this taxonomy answer different types of decision problems. These different decision problems still need to be better described and their added value for the decision maker should be more clearly highlighted.

## Acknowledgment

The authors wish to thank Yves de Smet for his advices. We especially acknowledge his contributions when the reflexions presented in this paper were still at an early stage.

## Appendixes

### A PROMETHEE Cluster

We show in this appendix that the PROMETHEE Cluster clustering procedure, defined by Figueira et al. [FSMB04] and slightly modified here to make it deterministic, is not criteria-dependant.

#### A.1 Definition of the procedure

We first describe the (modified) PROMETHEE Cluster procedure.

##### Context

The PROMETHEE Cluster procedure, defined for a given number of clusters  $K$ , takes as input a set of actions  $A$  and a set of criterion  $F = \{g_1, \dots, g_n\}$  defined on  $A$ , and finds the partition  $\rho$  on  $A$  which minimises the badness criterion  $E$  measuring the partition heterogeneity. This section explains how this partition heterogeneity measure  $E(\rho)$  is computed.

A partition  $\rho$  is a set of clusters. A cluster, denoted by  $\mathcal{C}$ , is a set of actions.

We warn the reader about the fact that in the original paper, the authors define the PROMETHEE Cluster procedure with a heuristic trying to find the optimal partition (i.e., a K-means using the given deviation measure). As we need a deterministic procedure to apply the criteria-dependance property, we present here a variant which always finds the optimal procedure, and thus do not talk about the heuristic. To make this PROMETHEE Cluster variant deterministic, we also suppose it uses some algorithm to deterministically choose an optimal partition in case of ex-æquo. The way the procedure chooses between these

equal-qualities partitions is not important for the proof, as long as the deterministic rule is based solely on the distance measure defined in the procedure and not directly on the criteria values.

### Central profiles

Having a cluster  $\mathcal{C}$ , we note  $r_{\mathcal{C}}$  the central profile of that cluster. Its evaluation values on each criterion  $g$  are computed as the average of the evaluations, on the same criterion, of the actions belonging to the same cluster, as follows:

$$g(r_{\mathcal{C}}) = \frac{1}{|\mathcal{C}|} \sum_{a \in \mathcal{C}} g(a), \forall g \text{ (where } |\mathcal{C}| \text{ is the number of actions in } \mathcal{C}\text{)}.$$

For a given partition  $\rho$ , we note  $\mathbf{R}$  the set of all central profiles of that partition:  $\mathbf{R} = \{r_{\mathcal{C}}, \forall \mathcal{C} \in \rho\}$ . We have  $|\mathbf{R}| = K$ .

### Preference intensities

To each criterion  $g$  is associated a preference function  $f_g : \mathbb{R} \rightarrow [0, 1]$ , used to compute the preference intensity  $\mathbf{P}_g$ ,  $\mathbf{P}_g(x, y)$  measuring how  $x$  is preferred to  $y$ .

$$\forall g \in \mathbf{F}, \mathbf{P}_g : \text{AUR} \times \text{AUR} \rightarrow [0, 1] \mid \forall x, y \in \text{AUR} : \mathbf{P}_g(x, y) = f_g(g(x) - g(y)).$$

### Net flow functions

The per criteria net flow is computed using two different formulas.

The per criteria net flow  $\phi_g^{\mathbf{R}}$ , for a criterion  $g$  and an object  $a \in \mathbf{A}$ , measures to which extent  $a$  is preferred over a set of reference objects, from the point of view of  $g$ .

$$\phi_g^{\mathbf{R}} : \mathbf{A} \rightarrow [0, 1] \mid \forall a \in \mathbf{A} : \phi_g^{\mathbf{R}}(a) = \frac{1}{|\mathbf{R}|} \sum_{r^* \in \mathbf{R}} (\mathbf{P}_g(a, r^*) - \mathbf{P}_g(r^*, a)).$$

The per criterion net flow of an reference action  $r \in \mathbf{R}$  is defined similarly.

$$\phi_g : \mathbf{R} \rightarrow [0, 1] \mid \forall r \in \mathbf{R} : \phi_g(r) = \frac{1}{|\mathbf{R}| - 1} \sum_{r^* \in \mathbf{R} \setminus \{r\}} (\mathbf{P}_g(r, r^*) - \mathbf{P}_g(r^*, r)).$$

The authors do not describe the case where  $|\mathbf{R}| = 1$ . We will thus consider that  $|\mathbf{R}| > 1$ .

## Deviation measure

The per criterion net flow are used to compute the **deviation** of an action compared to a reference profile:

$$\forall a \in A, r \in R : d(a, r) = \sqrt[p]{\sum_{g \in F} \frac{w_g}{W} |\phi_g^R(a) - \phi_g(r)|^p}.$$

## Heterogeneity measure

We can now define the heterogeneity measure:  $E(\rho) = \sum_{C \in \rho} \sum_{a \in C} d(a, r_C)$ . The optimal partition  $\rho$  is the one minimizing that measure (recall that we assume the procedure has a deterministic way of choosing between ex-æquos).

## A.2 PROMETHEE-Cluster is not criteria-dependent

Let us show that  $\forall A, F, g^* : f(A, F) = f(A, F')$ , with  $F' = INV(F, g^*)$ . This proves that the procedure is not criteria-dependant.

To show that the procedure's output is equal, it is sufficient to show that the distance measure they use are equal. Let us write  $d', \phi_g^{R'}, \phi_g', \mathbf{P}'_g$  the distance measure, the net flow on the action set, the net flow on the reference profiles set, and the preference function computed from the input  $F'$ . We thus have to show that  $d(a, r)$  and  $d'(a, r)$  are equal  $\forall F, g^*, a, r$ . To do this we need two intermediary results.

First, we have,  $\forall x, y \in A \cup R$  :

$$\mathbf{P}'_g(x, y) = \begin{cases} f_g(g(x) - g(y)) = \mathbf{P}_g(x, y) & \forall g \neq g^*, \\ f_g(-g(x) - (-g(y))) = f_g(g(y) - g(x)) = \mathbf{P}_g(y, x) & \text{for } g = g^* \end{cases}.$$

Thus,  $\mathbf{P}'_{g^*}(x, y) - \mathbf{P}'_{g^*}(y, x) = \mathbf{P}_{g^*}(y, x) - \mathbf{P}_{g^*}(x, y) = -(\mathbf{P}_{g^*}(x, y) - \mathbf{P}_{g^*}(y, x))$ .

Using that we can deduce a second result.

$\forall a \in A, r \in R, g \in F$ :  $|\phi_g^{R'}(a) - \phi_g'(r)| = |\phi_g^R(a) - \phi_g(r)|$ . This is trivially true when  $g \neq g^*$ . When  $g = g^*$ , we have:

$$\begin{aligned}
\left| \phi_{g^*}^{\mathbb{R}'}(a) - \phi_{g^*}'(r) \right| &= \left| \frac{1}{|\mathbb{R}|} \sum_{r^* \in \mathbb{R}} \left( \mathbf{P}'_{g^*}(a, r^*) - \mathbf{P}'_{g^*}(r^*, a) \right) \right. \\
&\quad \left. - \frac{1}{|\mathbb{R}| - 1} \sum_{r^* \in \mathbb{R} \setminus \{r\}} \left( \mathbf{P}'_{g^*}(r, r^*) - \mathbf{P}'_{g^*}(r^*, r) \right) \right| \\
&= \left| \frac{1}{|\mathbb{R}|} \sum_{r^* \in \mathbb{R}} - \left( \mathbf{P}_{g^*}(a, r^*) - \mathbf{P}_{g^*}(r^*, a) \right) \right. \\
&\quad \left. - \frac{1}{|\mathbb{R}| - 1} \sum_{r^* \in \mathbb{R} \setminus \{r\}} - \left( \mathbf{P}_{g^*}(r, r^*) - \mathbf{P}_{g^*}(r^*, r) \right) \right| \\
&= \left| \frac{1}{|\mathbb{R}|} \sum_{r^* \in \mathbb{R}} \left( \mathbf{P}_{g^*}(a, r^*) - \mathbf{P}_{g^*}(r^*, a) \right) \right. \\
&\quad \left. - \frac{1}{|\mathbb{R}| - 1} \sum_{r^* \in \mathbb{R} \setminus \{r\}} \left( \mathbf{P}_{g^*}(r, r^*) - \mathbf{P}_{g^*}(r^*, r) \right) \right| \\
&= \left| \phi_{g^*}^{\mathbb{R}}(a) - \phi_{g^*}(r) \right|.
\end{aligned}$$

$$\begin{aligned}
\text{And finally: } d'(a, r) &= \sqrt[p]{\sum_{g \in \mathbb{F}} \frac{w_g}{W} |\phi_{g^*}^{\mathbb{R}'}(a) - \phi_{g^*}'(r)|^p} \\
&= \sqrt[p]{\sum_{g \in \mathbb{F}} \frac{w_g}{W} |\phi_{g^*}^{\mathbb{R}}(a) - \phi_{g^*}(r)|^p} = d(a, r).
\end{aligned}$$

## B MC K-means

We analyze in this appendix the procedure, that we call ‘‘MC-K-means’’, defined by De Smet and Guzmàn [SG04]. After formally presenting the procedure, we proof by an example that the procedure is criteria-dependent.

### B.1 Definition of the procedure

The MC-K-means procedure assumes that a preference structure  $(P, I, J)$  has been built on the set of actions  $\mathcal{A} = \{a, b, \dots\}$ . A preference structure is a triplet of binary relations, where  $P$  models strict preference,  $I$  indifference and  $J$  incomparability. Such a preference structure can be constructed by using a multicriteria decision aid method.

Each action  $a \in \mathcal{A}$  is represented by the sets  $P^+(a)$ ,  $P^-(a)$ ,  $I(a)$  and  $J(a)$  containing the actions which are preferred by  $a$ , which prefer  $a$ , which are indifferent with  $a$  and which are incomparable with  $a$ :

$$\begin{aligned}
P^+(a) &= \{b \in \mathcal{A} : aPb\}. \\
P^-(a) &= \{b \in \mathcal{A} : bPa\}. \\
I(a) &= \{b \in \mathcal{A} : aIb\}.
\end{aligned}$$

$$J(a) = \{b \in \mathcal{A} : aJb\}.$$

A distance between any two action  $a, b \in \mathcal{A}$  is defined as follows:

$$d(a, b) = 1 - \frac{|P^+(a) \cap P^+(b)| + |P^-(a) \cap P^-(b)| + |I(a) \cap I(b)| + |J(a) \cap J(b)|}{n}.$$

This distance function measures the degree to which the two actions are similar with respect to the other actions. A particularity of this distance is that it does not only depend on the the two actions  $a$  and  $b$ , but on the whole set of actions  $\mathcal{A}$ .

Given a subset of actions  $\mathcal{A}' \subseteq \mathcal{A}$ , the sets  $P^+(r)$ ,  $P^-(r)$ ,  $I(r)$ , and  $J(r)$  of a centroid  $r$  of  $\mathcal{A}'$  are such that:

$$\begin{aligned} a \in P^+(r) & \text{ if } |\{b \in \mathcal{A}' : bPa\}| \\ & \geq \max\{|\{b \in \mathcal{A}' : aPb\}|, |\{b \in \mathcal{A}' : aIb\}|, |\{b \in \mathcal{A}' : aJb\}|\}. \end{aligned}$$

$$\begin{aligned} a \in P^-(r) & \text{ if } |\{b \in \mathcal{A}' : aPb\}| \\ & \geq \max\{|\{b \in \mathcal{A}' : bPa\}|, |\{b \in \mathcal{A}' : aIb\}|, |\{b \in \mathcal{A}' : aJb\}|\}. \end{aligned}$$

$$\begin{aligned} a \in I(r) & \text{ if } |\{b \in \mathcal{A}' : aIa\}| \\ & \geq \max\{|\{b \in \mathcal{A}' : aPb\}|, |\{b \in \mathcal{A}' : bPa\}|, |\{b \in \mathcal{A}' : aJb\}|\}. \end{aligned}$$

$$\begin{aligned} a \in J(r) & \text{ if } |\{b \in \mathcal{A}' : aJa\}| \\ & \geq \max\{|\{b \in \mathcal{A}' : aPb\}|, |\{b \in \mathcal{A}' : bPa\}|, |\{b \in \mathcal{A}' : aIb\}|\}. \end{aligned}$$

Furthermore, any action  $a \in \mathcal{A}$  can only belong to one of these four sets. If an action could be assigned to more than one set, one of the possibilities is chosen randomly.

The algorithm then follows a traditional K-means scheme, but we do not detail it. We assume that the procedure always chooses one of the partitions minimizing the average distances within each cluster (which one is chosen when there are several possibilities is not important for the proof).

## B.2 MC K-means is criteria-dependent

We illustrate the procedure on an example with 4 actions which can be found in Figure 3. We use the dominance relation in order to build the  $(P, I, J)$  structure. That is why we have the following sets:

	$P^+(\cdot)$	$P^-(\cdot)$	$I(\cdot)$	$J(\cdot)$
$a$		$b, c, d$	$a$	
$b$	$a$	$d$	$b$	$c$
$c$	$a$	$d$	$c$	$b$
$d$	$a, b, c$		$d$	



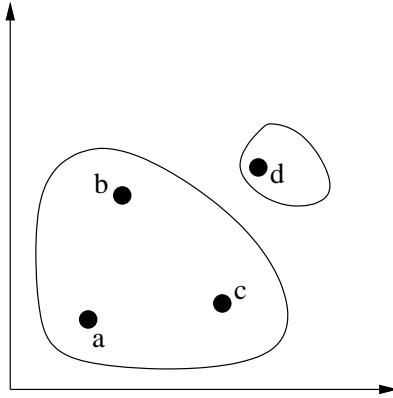


Figure 3: A 2-Cluster solution with four actions.

If we are looking for two clusters, then the following solution minimizes the average distance within the clusters:

Cluster	$P^+(r)$	$P^-(r)$	$I(r)$	$J(r)$	average distance
$\{a, b, c\}$	$a$	$d$	$b$	$c$	$5/12$
$\{d\}$	$a, b, c$		$d$		0

The distances between the actions of the first cluster and the centroid are  $d(a, r) = \frac{3}{4}$ ,  $d(b, r) = 0$  and  $d(c, r) = \frac{2}{4}$ , which leads to an average distance of  $\frac{5}{12}$ . In the second clusters, the action  $d$  coincides with the centroid and consequently the average distance in that cluster is 0. For this example, another optimal solution with 2 clusters would be  $\{a\}$  and  $\{b, c, d\}$ . In order to check for criteria-dependency, we must assume that the procedure deterministically outputs one of these two partitions. Whatever the tie-breaking mechanism used to chose between these two optimal solutions, we will now see that the procedure is criteria-dependent.

Let us now reverse the criterion on the vertical axis (see Figure 4). The preference structures now change accordingly:

	$P^+(\cdot)$	$P^-(\cdot)$	$I(\cdot)$	$J(\cdot)$
$a$			$a$	$d, b, c$
$b$		$c$	$b$	$a, d$
$c$	$b$		$c$	$a, d$
$d$			$d$	$a, b, c$

If we are looking for two clusters, then the only solution which minimizes the average distance within the clusters is:

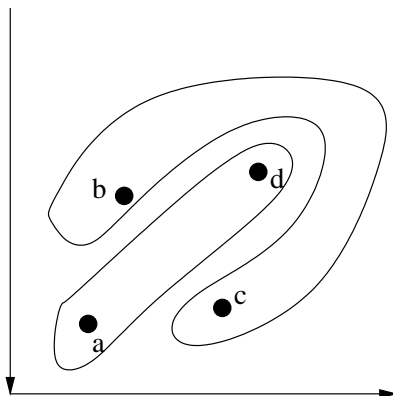


Figure 4: A 2-Cluster solution with four actions.

Cluster	$P^+(r)$	$P^-(r)$	$I(r)$	$J(r)$	average distance
$\{a, d\}$			$a$	$d, b, c$	$1/4$
$\{b, c\}$		$c$	$b$	$a, d$	$1/4$

Hence this solution is different from the two optimal solutions previously obtained. This illustrates the fact that the procedure is criteria-dependent, whatever the tie-breaking mechanism used to discriminate between multiple optimal solutions.

## References

- [Bis02] R. Bisdorff. Electre-like clustering from a pairwise fuzzy proximity index. *European Journal of Operational Research*, 138:320–331, 2002.
- [BVM86] J.P. Brans, Ph. Vincke, and B. Mareschal. How to select and how to rank projects: the promethee method. *European Journal of Operational Research*, 24:228–238, 1986.
- [DBF05] P. Doreian, V. Batagelj, and A. Ferligoj. *Generalized Blockmodeling*. Cambridge University Press, 2005.
- [DHS01] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern classification*. Wiley, New York, NY, USA, second edition, 2001. ISBN 0-471-05669-3. xx + 654 pp. LCCN Q327 .D83 2001.
- [ELL01] B. S. Everitt, S. Landau, and M. Leese. *Cluster Analysis*. A Hodder Arnold Publication, 2001.
- [FSMB04] J.R. Figueira, Y. De Smet, B. Mareschal, and J.-P. Brans. Mcda methods for sorting and clustering problems: Promethee tri and promethee cluster. Technical Report IS-MG 2004/02, Université Libre de Bruxelles/SMG, 2004.

- [JMF99] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, sep 1999. URL <http://citeseer.ist.psu.edu/jain99data.html>.
- [NL07] Ph. Nemery and C. Lamboray. Flowsort: a flow-based sorting method based on limiting or central profiles. In *Proceedings of the ORP3 conference*. Guimaraes, Portugal, 2007.
- [NS05] Ph. Nemery and Y. De Smet. Multicriteria ordered clustering. Technical Report TR/SMG/2005-003, Université Libre de Bruxelles/SMG, 2005.
- [PV97] M. PirLOT and Ph. Vincke. *Semi Orders*. Kluwer Academic, Dordrecht, 1997.
- [RB93] B. Roy and D. Bouyssou. *Aide Multicritère à la Décision : Méthodes et Cas*. Production et Techniques Quantitatives Appliquées à la Gestion (Collection Gestion). Economica, 49, rue Héricart - 75015 Paris, 1993.
- [Roy85] B. Roy. *Méthodologie Multicritère d'Aide à la Décision*. Production et Techniques Quantitatives Appliquées à la Gestion (Collection Gestion). Economica, 49, rue Héricart - 75015 Paris, 1985.
- [RV85] M. Roubens and Ph. Vincke. *Preference Modeling*. Springer Verlag, 1985.
- [SG04] Y. De Smet and L. Montano Guzmán. Towards multicriteria clustering: An extension of the k-means algorithm. *European Journal of Operational Research*, 158(2):390–398, oct 2004.
- [Yu92] W. Yu. Electre tri: Aspects méthodologiques et manuel d'utilisation. *Cahiers du Lamsade*, 74, 1992.
- [ZD02a] C. Zopounidis and M. Doumpos. Business failure predictions using the utadis multicriteria analysis method. *Journal of Operational Research Society*, 50:1138–1148, 2002.
- [ZD02b] C. Zopounidis and M. Doumpos. Multicriteria classification and sorting methods: A literature review. *European Journal of Operational Research*, 138:229–246, apr 2002.
- [Zop02] C. Zopounidis. MCDA methodologies for classification and sorting. *European Journal of Operational Research*, 138:227–228, 2002.