



HAL
open science

Inferring a hierarchical majority-rule sorting model

Alexandru-Liviu Olteanu, Patrick Meyer

► **To cite this version:**

Alexandru-Liviu Olteanu, Patrick Meyer. Inferring a hierarchical majority-rule sorting model. Computers and Operations Research, 2022, 146, pp.105888. 10.1016/j.cor.2022.105888 . hal-03702588

HAL Id: hal-03702588

<https://hal.science/hal-03702588v1>

Submitted on 22 Jul 2024

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.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Inferring a hierarchical majority-rule sorting model

Abstract

Real decision problems often involve a multitude of sub-problems that are intertwined and that need to be solved together in order to provide a global recommendation. A hierarchical structure may be used to represent such problems and we propose an indirect elicitation approach when each underlying problem can be modeled as a majority-rule sorting problem. We limit our study to a two level hierarchy structure, where the bounding profiles of the bottom layer are provided while those of the top layer need to be inferred. We propose a mathematical program to infer these profiles as well as the criteria weights and majority thresholds of all considered models. We test this program on generated benchmarks in order to evaluate its performance and find its limitations. The approach may furthermore be easily extended to more complex hierarchical structures.

Keywords: multi-criteria decision aiding, majority-rule sorting, hierarchy of models, model inference, mathematical programming

1. Introduction

Multi-criteria decision aiding (MCDA) is aimed at helping decision-makers (DM) deal with complex problems involving multiple, potentially conflicting, perspectives, or criteria, on the decision problem [1]. These problems generally involve one out of three types of decisions: choice, or selecting the best alternatives, ranking, or ordering alternatives from the best one to the worst, and sorting, or assigning alternatives into predefined ordered categories.

Usually, a decision aiding process is put in place [2], where, during the initial phases, the decision problem is structured and the set of criteria and alternatives are identified. Complex decision problems may involve multiple decision levels or, in other cases, criteria that are used to form broader, more general criteria. The latter has been formally studied as the Multiple Criteria Hierarchical Process [3] and adapted to a multitude of decision problems and preference models. The former, to our knowledge, has received less attention as it may be seen as a generalization of the latter. As such, tuning multiple preference models that form a hierarchy is still an open topic.

Providing the preference model parameters directly by the DM, or through a protocol with the help of an analyst, is generally difficult even when the problem does not hold a hierarchical structure. We therefore propose, in this

work, to indirectly infer the parameters of what we call a hierarchical majority-rule sorting model through the use of a mathematical programming approach.

We show that, using current computational resources, we are able to infer accurate models in a relatively short amount of time when a two-level hierarchy is considered, using only holistic information on the output of the sorting procedure. We also investigate the impact of erroneous information on the convergence of the proposed approach, which becomes marginal when sufficient information is provided in order to infer the preference model, requiring, however, a considerable increase in computational resources.

The paper is structured in the following way. Section 2 presents the state of the art on the integration of hierarchies within a decision process as well as on the indirect inference of preference models. Section 3 formally defines the proposed hierarchical majority-rule sorting approach and illustrates it using an intuitive example. In Section 4 we present and explain the mathematical program that will be used to infer the model from a set of assignment examples, also taking into account robustness considerations. Finally, Section 5 outlines the experimental protocol and its results before ending with several concluding remarks and perspectives for future work in Section 6.

2. State of the art

2.1. Integrating hierarchies of criteria into the decision process

MCDA focuses on helping DMs when alternatives evaluated on multiple, often conflicting criteria are considered [1]. In the majority of applications, criteria are considered to be on the same level and taken into account all at once, while the decision problem is considered to be atomic, not being comprised of multiple underlying decisions.

In certain cases, however, the criteria may form a hierarchical structure, where a criterion is divided into sub-criteria. An introduction of MCHP to ELECTRE methods may be found in [4], while the particular case of the ELECTRE-Tri-B method was covered in [3]. In the latter, the process aggregates sets of criteria in order to form new criteria using the ELECTRE-Tri-B sorting mechanism, which then in turn may be further aggregated into new criteria and so on, until reaching a top level node in this tree-like structure. The majority-rule sorting method [5], on which this article is focused, is a particular case of the ELECTRE-Tri-B method. The MCHP has also more recently been adapted to sorting problems under uncertainty for the evaluation of the operational maturity of research institutions [6]. The ranking method ELECTRE III has also been extended to incorporate a hierarchy of criteria in [7], with interactions between the hierarchy of criteria and imprecision with respect to their weights being covered in [8].

Decisions may also need to account for different scenarios or context variations. The most common way of dealing with such variations, when they affect the potential evaluations of an alternative, is to integrate them by using a top-down or bottom-up approach during the hierarchical construction of the set of

criteria [9]. The variations are therefore aggregated into a single evaluation, using for instance expert rules or other MCDA approaches for modeling the sub-problem of constructing the criterion. In some cases, it may be important for the DM to take into account these variations across all criteria differently from one context to another. Few proposals have considered multi-criteria decision problems and multiple contexts at the same time. We mention here the work of Zheng et al. [10] proposing a methodology coupling context-aware recommender systems and multi-criteria recommender systems within a case study linked to education and learning. Frini et al. [11] propose a new MCDA outranking methodology called MUPOM which takes into account multiple periods of time. Breaking time into periods allows to consider time as a context which influences variations in the evaluations of alternatives. This work analyzes multiple time periods for sustainable development of forests and takes into account time-specific factors such as continuity. For each time period, outranking relations are computed between any pair of alternatives aggregating multiple concordance and discordance indices for each criterion and time step. Each time period is then taken into account through the use of a distance measure that uses the previously constructed relations. Finally, the overall outranking relation is then exploited in order to provide a recommendation. Nevertheless, due to the complexity of the approach, the process of parameters elicitation may prove to be difficult from the perspective of a DM.

Problems involving multiple DMs may also lead to a hierarchical structure, this time in relation to a series of decisions that form the basis for higher level decisions. One of the first contributions dealing with the sorting problem in the context of group decision making is that of Dias and Clímaco [12], where the case of aggregating the output of each DMs ELECTRE-TRI model may be considered as a two-level hierarchical sorting model. Jabeur and Martel [13] argue against the use of the same type of model, the same set of criteria and alternatives evaluations as well as considering all DMs as equally important in the construction of the final recommendation. They propose a method where the DMs agree on the set of categories, criteria and evaluations of reference alternatives, but where individual models on potentially different alternatives evaluations are constructed for each DM and their outputs are later aggregated using relative DM importance coefficients. In [14, 15] robust ordinal regression was extended to deal with sorting problems and group decision making. The FlowSort method was also extended to a group decision making context in [16]. A recent and extensive study covering sorting methods in MCDA, including those dealing with group decision making may be found in [17].

2.2. Indirect inference of preference models

The indirect inference of a preference model generally seeks to overcome some of the difficulties associated with requiring the DM to properly understand and provide the parameters of a preference model [18]. Holistic judgements on the desired output of the MCDA method on a subset of the decision problem are hence provided by the DM and used by an analytical approach in order to infer the parameters of the preference model. This topic has received a lot of interest,

namely within the works of Jacquet-Lagrange and Siskos [19] in the Multiple Attribute Value Theory (MAVT; [9]) context and of Ngo The and Mousseau [20] with respect to outranking methods. These approaches determine the parameters of the considered preference models in one go, using, often large, sets of judgements provided by the DM. Incremental approaches, implying multiple and successive interactions with the DM, may also be found, such as [21, 22, 23, 24] in the MAVT context, to name a few. More recently, such approaches in an outranking context have also been proposed [25, 26].

Within the context of outranking-based sorting models, several works have been proposed for inferring the parameters of preference models, most being linked to the ELECTRE TRI method. In [27, 28, 20] it has been proposed to find the model parameters through the use of *assignment examples*. More specifically, the DM is initially asked to assign a few *well known* alternatives to the predefined categories, followed by the use of linear or non-linear programs in order to output the model parameters. More robust approaches compute for each alternative a range of possible categories to which they may be assigned when the parameters of the model are not completely determined [29, 12, 30]. Approaches that deal with inconsistencies in the assignments given by the DM have also been explored in [31, 32].

In most cases, approaches using mathematical programming techniques involving binary variables (e.g. in [5]) find the optimal solution, but require large amounts of computational resources and time, being able to handle only instances of reduced size. Sobrie et al. [33] have suggested to use a metaheuristic to learn the parameters of the MR-Sort model, which has been adapted and extended by Olteanu and Meyer [34] in order to also take into account veto thresholds. Population-based metaheuristics have also been proposed for MR-Sort models with coalitional veto [35], while a matheuristic approach (combining metaheuristics and mathematical programming) was proposed in [36] in order to infer an MR-Sort model able to deal with imprecise or missing evaluations. Finally, exact elicitation approaches for inferring MR-Sort models containing large performance differences, such as vetoes or dictators may be found in [37].

3. Hierarchical sorting problems

We consider extending the MR-Sort model [5, 33] to the case where the sorting problem can be decomposed into multiple sub-problems and where each sub-problem provides a partial aggregation of the evaluations of an alternative on a set of criteria. This may be needed for multiple types of applications which we will present hereafter, followed by a formal definition of what we call the HMR-Sort model, which stands for Hierarchical Majority-Rule Sorting.

3.1. Considered problems

We depict in Figure 1 an example of a more or less generic hierarchical structure for a sorting problem.

We observe a problem where each alternative a is evaluated on 9 criteria (g_1 through g_9), that may be found at the bottom of the figure. Furthermore,

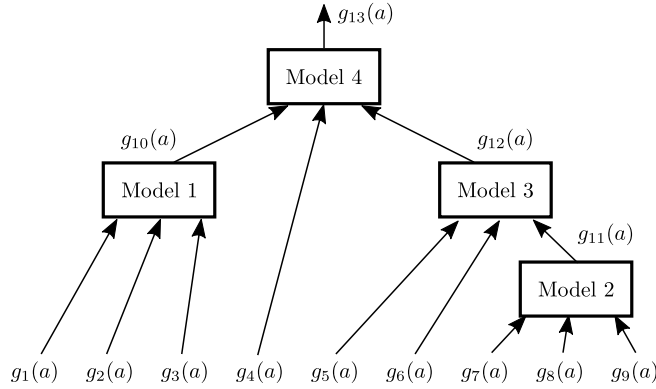


Figure 1: Example of a hierarchical sorting problem

4 models are used to provide partial classifications taking as input different subsets of criteria as well as the outputs of some of the other models. Each model may be seen as a sub-problem, with the top model providing the final recommendation. The first model uses the first three criteria as input in order to provide as output the g_{10} aggregated criterion. Similarly, the second model uses the last 3 criteria, while its output is regarded as an intermediate criterion and used together with criteria g_5 and g_6 as input by the third model. Finally, the intermediate criteria constructed by models 1 and 3, along with criterion g_4 , are used by the fourth model in order to provide the final classification of the alternative. We consider each model to correspond to a separate sorting problem with its own parameters.

Based on the number of models and how the criteria evaluations of the alternatives are shared by them, multiple scenarios may be identified. The most basic scenario corresponds to a single model taking into consideration all provided criteria. This represents a classical sorting problem. When multiple models may be found, the hierarchical structure may be seen as an acyclic rooted graph, with the model at the root providing the overall recommendation. When all models are only connected directly to the root model, then the problem corresponds to a two-level problem. Multi-level sorting problems may further contain models that are connected to other sorting models instead of the root model. In both cases, the underlying models may integrate potentially different DMs.

Among the two-level scenario, we can additionally distinguish several classes based on how the criteria are shared among the intermediate models:

- **Identical criteria**
This type of model considers that multiple DMs evaluate the same decision problem using the same criteria and the same evaluations however have different preferences and will therefore provide potentially different recommendations. A global DM then provides an overall recommendation only considering the recommendations of the underlying DMs;

- Partially identical criteria
This type of model is similar to the previous one except the underlying models do not necessarily consider the same sets of criteria. This may correspond to multiple DMs evaluating alternatives based on their field of expertise and thus using criteria that are pertinent only to them. Then, a global DM uses the recommendations of the underlying DMs in order to provide a global one;
- Disjoint criteria
This model considers that no criterion is shared between the underlying models.

The previous categories may further be separated based on how the alternatives are evaluated on the criteria of the intermediate models:

- Identical evaluations
In this case, if criteria are shared between models, a specific evaluation of an alternative on a given criterion will be identical on all models that consider this criterion;
- Contextual evaluations
In this case, even if a criterion is shared by multiple models, the evaluations of an alternative may differ from one model to another. This may be used for problems where an alternative is evaluated on the same criteria w.r.t. different contexts, such as, for instance, the productivity of a system based on different imposed workload scenarios.

A further distinction may be made when the criteria are shared among multiple models while the alternatives evaluations on them differ from one model to the next due to context differences. In this case, we may either have the same DM that is shared across all models therefore having the same preferences or different DMs with potentially different preferences. Such a situation may be used to model problems where an alternative is evaluated across different time periods or within different contexts using the same criteria, where the performance expectations of the DM stay the same.

3.2. Hierarchical majority-rule sorting model

We continue by formally defining the Hierarchical Majority-Rule Sorting model, or HMR-Sort.

Let us consider a finite set of alternatives A and a finite set of criteria indexes $J = \{1, \dots, m\}$. Each alternative $a \in A$ is evaluated on any criterion through a function $g_j : A \rightarrow \mathbb{R}$, where $g_j(a)$ ($j \in J$) denotes the performance of the alternative a on criterion g_j . We assume without loss of generality that the preference directions on each criterion are increasing. The alternatives are to be sorted into a set of k ordered categories $C = \{c_1, \dots, c_k\}$, with c_1 being the least preferred category and c_k the most preferred one.

Table 1: The mapping function γ for the illustrative example

$\gamma(i, j)$		j		
		1	2	3
i	1	1	2	3
	2	7	8	9
	3	5	6	11
	4	10	4	12

We also consider a decomposition of the decision problem into sub-problems, which we denote using a set of indexes $I = \{1, \dots, n\}$. All sub-problems correspond to sorting problems where MR-Sort models are used and are arranged in order to form a rooted acyclic graph structure. The model corresponding to the root gives the overall assignment for the sorting problem. We may refer to this model as the global model and to the other models as intermediate models. We additionally consider that the global model index is equal to n . Each model is used to assign an alternative to a category among a set of ordered categories $C^i = \{c_1^i, \dots, c_{k^i}^i\}$, $\forall i \in I$, with category c_1^i being the least preferred and category $c_{k^i}^i$ being the most preferred. Note that $C^n \equiv C$.

We denote the criteria for a sub-problem $i \in I$ using indexes $J^i \subseteq \{1, \dots, m+n\}$, where we denote $m^i = |J^i|$. The result of an MR-Sort model $i \in I$ induces what we call an intermediate evaluation of an alternative $a \in A$. In order to aid in standardizing the subsequent definitions, we use for these intermediate evaluations the same function g that is used to hold the evaluations of an alternative $a \in A$ on any criterion $j \in J$, except that we will denote with $g_{m+i}(a)$ the intermediate evaluation of a based on model $i \in I$. In order to identify the evaluation of a model $i \in I$ on a criterion $j \in J^i$ for an alternative $a \in A$, we use a mapping function $\gamma : \mathbb{N}^+ \times \mathbb{N}^+ \rightarrow \{1, \dots, m+n\}$, such that $g_{\gamma(i,j)}(a)$ retrieves this evaluation, be it real or intermediate.

An illustration of a mapping function for the example highlighted in Figure 1 may be found in Table 1.

Each MR-Sort model $i \in I$ from a hierarchical sorting problem contains a set of parameters such as :

- the bounding profiles $B^i = \{b_0^i, \dots, b_{k^i}^i\}$, where $b_h^i \geq b_{h-1}^i, \forall h \in \{1, \dots, k^i\}$;
- the criteria weights $w_j^i \in (0, 1), \forall j \in J^i$, where $\sum_{j \in J^i} w_j^i = 1$;
- the majority threshold $\lambda^i > 0.5$.

Each MR-Sort model $i \in I$ outputs in g_{m+i} the index $h \in \{1, \dots, k^i\}$ of the category to which it assigns an alternative $a \in A$. Two rules to assign an alternative to a class may be found in the literature, the pessimistic and the optimistic assignment rules, out of which the first is the most commonly used.

In this case, an alternative a is assigned to the highest possible category c_h^i such that a outranks the category's lower frontier b_{h-1}^i . Alternative a is said to outrank b_{h-1}^i if and only if there is a sufficient coalition of criteria supporting the assertion “ a is at least as good as b_{h-1}^i ”. More precisely, partial concordance indexes C_j^i are first defined to assess whether each criterion $g_{\gamma(i,j)}$ supports this statement:

$$\forall i, j, a, h \in I \times J^i \times A \times \{1, \dots, k+1\} : C_j^i(a, b_{h-1}^i) = \begin{cases} 1, & \text{if } g_{\gamma(i,j)}(a) \geq g_{\gamma(i,j)}(b_{h-1}^i), \\ 0, & \text{otherwise.} \end{cases}$$

A global concordance index, denoted as $C^i(a, b_{h-1}^i), \forall a \in A, h \in \{1, \dots, k+1\}$, is then defined as:

$$C^i(a, b_{h-1}^i) = \sum_{j \in J^i} w_j^i C_j^i(a, b_{h-1}^i). \quad (2)$$

The global concordance index is compared to the majority threshold λ^i and therefore a is said to outrank b_{h-1}^i , i.e. $a S b_{h-1}^i$ if $C^i(a, b_{h-1}^i) \geq \lambda^i$. If $C^i(a, b_{h-1}^i) < \lambda$, the support in favor of $a S b_{h-1}^i$ is not sufficient and so a does not outrank b_{h-1}^i , i.e. $a \not S b_{h-1}^i$. To summarize, $a \in c_h^i \iff a S b_{h-1}^i \wedge a \not S b_h^i$.

3.3. Illustrative example

Let us consider a potential application of the proposed model within an industrial context. The choice of new equipment in the manufacturing industry generally needs to take into account multiple criteria such as its initial cost, production capacity and operation safety. Constructing these criteria, however, is a difficult process and may hide a needed, more detailed, characterization of the equipment based on different use contexts. For example, when the initial investment is important and the production is known to fluctuate based on different foreseen market factors, the industrial actor might prefer to have a clear understanding of how the equipment will perform during these different contexts and base their decision on it. Dividing the problem of determining whether an equipment performs overall well may therefore benefit from splitting the problem into multiple sub-problems linked to each use context.

We consider, in this example, that a company seeks to invest in a new equipment for a specific production task. The initial investment costs are rather similar and therefore the investor wishes to select an equipment that can adapt to the market fluctuations and the subsequent production rhythms. Three such rhythms are identified:

- nominal production, as determined by the manufacturer of the equipment;
- reduced production, which is characterized by large fluctuations due to having fewer orders which arrive at a more or less unpredictable rate;
- intensive production corresponding to a rapid production rate in order to handle a larger than usual order volume.

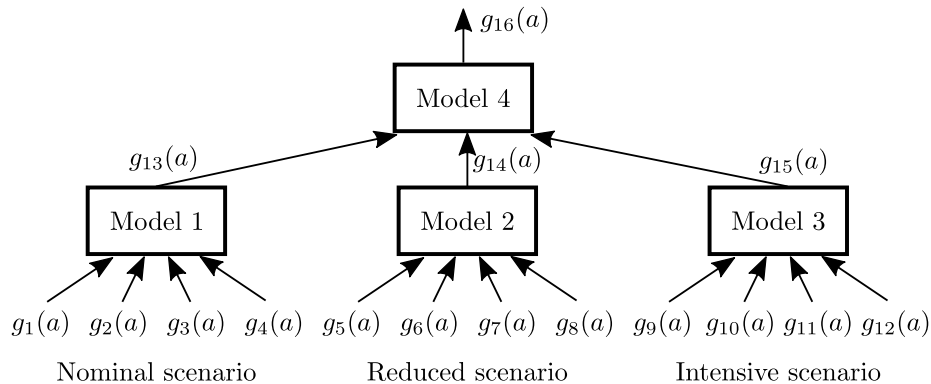


Figure 2: Example of hierarchical problem structure

For each of the three cases, the same four criteria will be used to evaluate the efficiency of the equipment:

- production rate as in units per hour, ranging between 0 and 20;
- operating cost, taking into account multiple factors such as energy consumption, consumables, maintenance, etc., and ranging between 0 and 60;
- required operators from 0 to 4;
- safety index, given as ordinal levels from D to A^{++} , with the former being the worst and the latter being the best.

However, as each alternative will be evaluated in different working conditions on these criteria, the criteria will naturally depend on these contexts and therefore effectively be different criteria. As such, each alternative will hold 12 criteria evaluations.

It is clear that an investor would prefer an equipment that maximizes the production rate, minimizes the operation costs, minimizes the needed number of operators and maximizes safety, for all of the three previously defined scenarios. An equipment that performs well across all of these criteria is very unlikely to exist, hence a compromise solution needs to be found.

We define a set of three categories: Good (G), Medium (M) and Bad (B). Since taking into account all four criteria for all three scenarios may prove difficult from the perspective of the DM, we divide the problem by first evaluating the equipment based on each of the three scenarios and then using these evaluations in order to determine its overall performance.

We present in Figure 2 the structure of the problem and in Table 2 the parameters of the four models that constitute the HMR-Sort model for this problem. We assume these parameters were elicited from the DM.

Figure 3 illustrates the assignment of an alternative x , which is evaluated on the four criteria based on each of the three production rhythms. The alternative

				g_{13}	g_{14}	g_{15}								
				b_2^4	3	3	3							
				b_1^4	2	3	2							
				w^4	0.2	0.4	0.4							
				λ^4	0.6									
(a) Model 4 (global)														
g_1				g_5				g_9						
g_2				g_6				g_{10}						
g_3				g_7				g_{11}						
g_4				g_8				g_{12}						
b_2^1	10	30	2	A^+	b_2^2	7	20	1	A^+	b_2^3	20	50	2	A^+
b_1^1	8	50	3	B^+	b_1^2	5	30	1	A	b_1^3	15	60	3	B^-
w^1	0.3	0.2	0.3	0.2	w^2	0.1	0.3	0.3	0.3	w^3	0.4	0.25	0.1	0.25
λ^1	0.7				λ^2	0.7				λ^3	0.65			
(b) Model 1 (nominal)				(c) Model 2 (reduced)				(d) Model 3 (intensive)						

Table 2: Example of HMR-Sort model parameters

evaluations (on the original criteria as well as the intermediate ones) are depicted in black.

We observe that, for a nominal context, x is at least as good as profile b_1^1 on all criteria and it is at least as good as profile b_2^1 w.r.t. g_1 (production capacity), g_3 (required number of operators) and g_4 (safety). This means that x outranks both profiles (based on the criteria weights and majority threshold of Model 1) and is therefore considered as a good equipment for this context. The output of this model is therefore equal to 3, the index of the Good category.

Similarly, x is considered to be good for a reduced production context, as it is at least as good as both profiles on all criteria except the first one.

Finally, for the intensive production context, x is at least as good as b_1^3 on all criteria except the second, therefore it outranks this profile, however it is at least as good as b_2^3 on only the last criterion (safety). In other words, x is only considered to have a medium performance for this context. These assignments give x an intermediate evaluation of $(3, 3, 2)$, which is then used by the last model. These evaluations make alternative x outrank profile b_1^4 on all criteria (which correspond to the overall performance for each production rhythm) and also outrank profile b_2^4 for the first two. Hence, alternative x is considered to be overall good.

4. Inferring the parameters of the model

The process of eliciting an HMR-Sort model may be divided in two, as is often the case in literature: direct elicitation or indirect elicitation. The first considers that we have access to all the DMs involved in the decision process and that they are able to provide the exact values of the parameters of each MR-Sort model. We recall that for each MR-Sort model we have several bounding

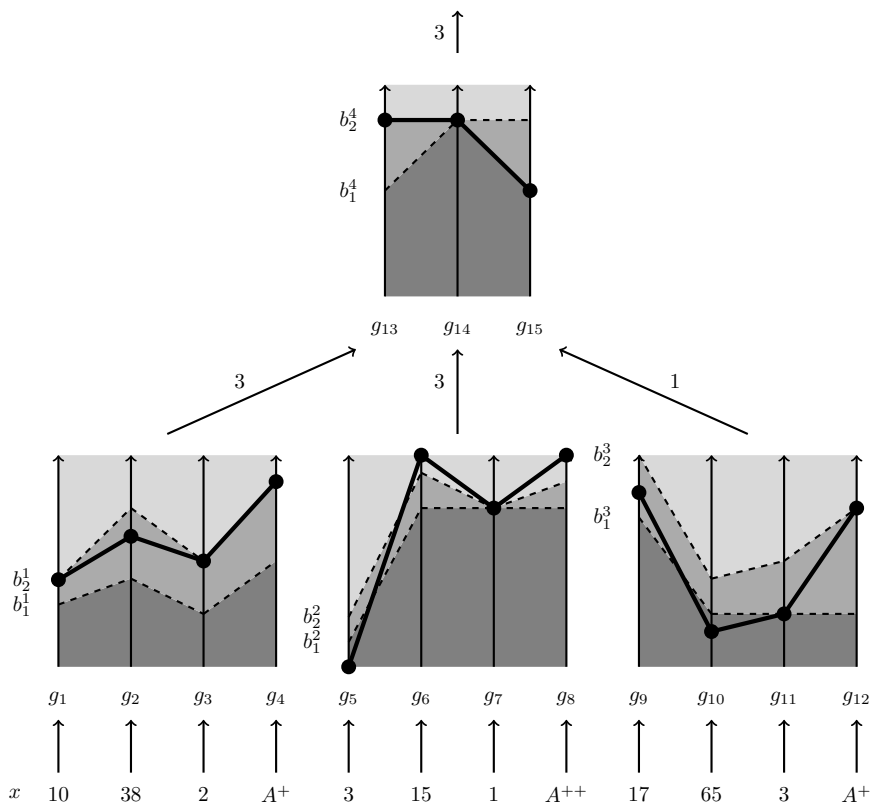


Figure 3: Example of HMR-Sort assignment

profiles, criteria weights and a majority threshold. The second approach consists in inferring the parameters of each MR-Sort model from holistic information of the DMs on the expected output, in other words the assignments of alternatives to their corresponding categories, also called assignment examples.

We focus our attention on the indirect elicitation of an HMR-Sort model. When all of the DMs are available, this process does not differ at all from an indirect elicitation of MR-Sort models, except that multiple models need to be inferred at the same time.

Interacting with multiple DMs, however, may prove difficult or even impossible in practice. It may also be of particular interest to reconstruct a clear and easy to understand model when information on partial decisions (the output of the intermediate models) is difficult to come by. We consider therefore the case of inferring an entire HMR-Sort model using the assignment examples of the top level model without having access to the intermediate classifications.

4.1. Principle

We consider HMR-Sort models defined by:

- a set of m criteria with indexes $J = \{1, \dots, m\}$;
- a set of k ordered categories $C = \{c_1, \dots, c_k\}$;
- a set of n subproblems with indexes $I = \{1, \dots, n\}$, each defined by:
 - a set of m^i criteria with indexes $J^i = \{1, \dots, m^i\}$;
 - a set of k^i ordered categories $C^i = \{c_1^i, \dots, c_{k^i}^i\}$;
- a two-level hierarchy structure where:
 - the intermediate problems correspond to models $i \in \{1, \dots, n - 1\}$ with their criteria being mapped to the set of real criteria $\gamma(i, j) = j, \forall j \in \{1, \dots, m\}$;
 - the top problem corresponds to model n with its criteria being mapped to the output of the intermediate problems $\gamma(n, i) = m + i, \forall i \in \{1, \dots, n - 1\}$;

We propose an indirect elicitation procedure in which a decision maker provides a list assignment examples in the form of a set of alternatives \mathcal{A} and a list K of their assignments to the set of categories C . We further consider that the boundary profiles of the intermediate MR-Sort models ($i \in \{1, \dots, n - 1\}$) have been elicited directly. This is motivated by the fact that considering each criterion separately from the rest and identifying the thresholds that correspond to ordinal performance levels is cognitively easy for a DM while evaluating the relative importance of criteria and the required overall levels of importance needed to model an outranking relation is significantly more difficult.

4.2. Mathematical Program for finding a feasible model

In this section, we define a mathematical formulation for learning, from a given set of assignment examples provided by the DM, the criteria weights $w_j^i, \forall i \in \{1, \dots, n\}, j \in \{1, \dots, m^i\}$ and majority thresholds $\lambda^i, \forall i \in \{1, \dots, n\}$ of all MR-Sort models of the previously defined HMR-Sort model as well as the bounding profiles $B^n = \{b_0^n, \dots, b_{k^n}^n\}$ of the top MR-Sort model.

The parameters that are considered as data in the proposed mathematical model are provided in Table 3.

A	the set of alternatives (N in total)
K	the category assignments of the alternatives in the form of category indices (N in total), with $K(a) \in \{1, \dots, k\}, \forall a \in A$
I	the set of model indices (n in total)
J^i	the set of criteria indices for each model $i \in I$ (m^i in total)
k^i	the number of categories for each model $i \in I$
G	the alternatives evaluations, or performance table, given as a matrix of size $N \times M$ (with $g_{a,j} \in [0, 1]$ containing the evaluation of alternative $a \in A$ on criterion $j \in J$)
$b_{h,j}^i$	the performance of the bounding profiles of all models except the top one $\forall i \in I \setminus \{n\}, \forall j \in J^i, \forall h \in \{1, \dots, k^i - 1\}$
γ_j^i	the criteria mapping for each model $i \in I$ and their corresponding criteria $j \in J^i$; this corresponds to the either the index of the real criterion $j \in J$ or the index of the intermediate criterion $i \in I \setminus \{n\}$
ϵ	a small positive constant used to model strict inequalities

Table 3: Parameters of the mathematical model.

We consider, without loss of generality, that the criteria evaluations scales are defined on the unit interval and that larger values are preferred to smaller ones. In order to apply the proposed model to real problem instances, a simple transformation of the criteria scales is required.

The variables considered in the mathematical program correspond to the HMR-Sort parameters that are to be inferred (i.e, criteria weights w_j^i and majority thresholds λ^i for all models i as well as bounding profiles b_h^n for the top model), and additional “technical” variables which are necessary to formulate the constraints. These variables are listed in Table 4.

The Mixed-Integer Linear Programming formulation for inferring an HMR-Sort model does not have any objective function, as we are interested only in finding a feasible solution according to the following constraints:

$$1 \geq \lambda_i \geq 0.5 + \epsilon \quad \forall j \in J^i, \forall i \in I \quad (3)$$

$$\sum_{j \in J^i} w_j^i = 1, \text{ with } w_j^i \geq \epsilon \quad \forall j \in J^i, \forall i \in I \quad (4)$$

$$1 \geq b_{h+1,j}^n \geq b_{h,j}^n \geq 0 \quad \forall j \in J^n, \forall h \in \{1, \dots, k^i - 1\} \quad (5)$$

w_j^i	<i>continuous</i>	:	the criteria weights of each model, $\forall i \in I, \forall j \in J^i$
λ^i	<i>continuous</i>	:	the majority thresholds of each model, $\forall i \in I$
$b_{h,j}^n$	<i>continuous</i>	:	the performance of the bounding profiles of the top model $\forall j \in J^n, \forall h \in \{1, \dots, k^n - 1\}$
$s_{a,h}^i$	<i>binary</i>	:	1 if alternative a outranks profile b_h^i , and 0 otherwise $\forall a \in A, \forall i \in I \setminus \{1\}, \forall h \in \{1, \dots, k^i - 1\}$
$\delta_{a,j}^+$	<i>binary</i>	:	1 if alternative a outranks the upper profile of its assigned category $K(a)$ of the top model on criterion $j \in J^n$, i.e. $b_{K(a),j}^n$, and 0 otherwise $\forall a \in A, \forall j \in J^n$
$\delta_{a,j}^-$	<i>binary</i>	:	1 if alternative a outranks the lower profile of its assigned category $K(a)$ of the top model on criterion $j \in J^n$, i.e. $b_{K(a)-1,j}^n$, and 0 otherwise $\forall a \in A, \forall j \in J^n$
$\omega_{a,j}^+$	<i>continuous</i>	:	equal to w_j^n if $\delta_{a,j}^+ = 1$ and to 0 otherwise, $\forall a \in A, \forall j \in J^n$
$\omega_{a,j}^-$	<i>continuous</i>	:	equal to w_j^n if $\delta_{a,j}^- = 1$ and to 0 otherwise, $\forall a \in A, \forall j \in J^n$

Table 4: Variables of the mathematical model.

$$\sum_{\substack{j \in J^i \\ g_{a,\gamma_j^i} \geq b_{h,j}^i}} w_j^i \geq \lambda^i + s_{a,h}^i - 1 \quad \forall a \in A, \forall i \in I \setminus \{n\}, \forall h \in \{1, \dots, k^i - 1\} \quad (6)$$

$$\sum_{\substack{j \in J^i \\ g_{a,\gamma_j^i} \geq b_{h,j}^i}} w_j^i + \epsilon \leq \lambda^i + s_{a,h}^i \cdot (1 + \epsilon) \quad \forall a \in A, \forall i \in I \setminus \{n\}, \forall h \in \{1, \dots, k^i - 1\} \quad (7)$$

$$\sum_{h=\{1, \dots, k^i - 1\}} \frac{s_{a,h}^{\gamma_j^n}}{k^i - 1} - b_{K(a)-1,j}^n \geq \delta_{a,j}^- - 1 \quad \forall a \in A, \forall j \in J^n \text{ if } K(a) > 1 \quad (8)$$

$$\sum_{h=\{1, \dots, k^i - 1\}} \frac{s_{a,h}^{\gamma_j^n}}{k^i - 1} - b_{K(a)-1,j}^n + \epsilon \leq \delta_{a,j}^- \cdot (1 + \epsilon) \quad \forall a \in A, \forall j \in J^n \text{ if } K(a) > 1 \quad (9)$$

$$\sum_{h=\{1, \dots, k^i - 1\}} \frac{s_{a,h}^{\gamma_j^n}}{k^i - 1} - b_{K(a),j}^n \geq \delta_{a,j}^+ - 1 \quad \forall a \in A, \forall j \in J^n \text{ if } K(a) < k^n \quad (10)$$

$$\sum_{h=\{1, \dots, k^i - 1\}} \frac{s_{a,h}^{\gamma_j^n}}{k^i - 1} - b_{K(a),j}^n + \epsilon \leq \delta_{a,j}^+ \cdot (1 + \epsilon) \quad \forall a \in A, \forall j \in J^n \text{ if } K(a) < k^n \quad (11)$$

$$w_j^n \geq \omega_{a,j}^- \geq 0 \quad \forall a \in A, \forall j \in J^n \quad (12)$$

$$\delta_{a,j}^- \geq \omega_{a,j}^- \geq \delta_{a,j}^- + w_j^n - 1 \quad \forall a \in A, \forall j \in J^n \quad (13)$$

$$w_j^n \geq \omega_{a,j}^+ \geq 0 \quad \forall a \in A, \forall j \in J^n \quad (14)$$

$$\delta_{a,j}^+ \geq \omega_{a,j}^+ \geq \delta_{a,j}^+ + w_j^n - 1 \quad \forall a \in A, \forall j \in J^n \quad (15)$$

$$\sum_{j \in J^n} \omega_{a,j}^- \geq \lambda^n \quad \forall a \in A \quad (16)$$

$$\sum_{j \in J^n} \omega_{a,j}^+ + \epsilon \leq \lambda^n \quad \forall a \in A \quad (17)$$

$$(18)$$

Constraint (3) limits the majority thresholds of all models to an interval where the coalition of criteria in favor of an outranking corresponds to a qualified majority.

Constraint (4) is needed to normalize the criteria weights. A non-zero lower-bound is also added in order to have all the criteria play a role in the pairwise comparisons of alternatives.

Constraint (5) is used to bound the profiles evaluations to the unit interval and to ensure the dominance structure on the set of profiles. We assume here, without loss of generality, that all criteria scales are in the unit interval and that larger evaluations are preferred to lower ones.

Constraints (6) and (7) are used to model the outranking relations between an alternative $a \in A$ and the profiles b_h^i for any model $i \in I \setminus \{n\}$, $\forall h \in \{1, \dots, k^i - 1\}$. The bottom profile of the lowest category and the top profile of the highest category are not considered in order to reduce the number of unnecessary variables and constraints. Since both the alternative and profiles evaluations are known, we take the sum of the weights where the alternative is at least as good as the profile and compare it to the majority threshold λ^i . If the former is larger or equal to the latter then $s_{a,h}^i$ becomes 1, otherwise $s_{a,h}^i$ becomes 0. $s_{a,h}^i$ corresponds to the global concordance relation between a and b_h^i . Also note that we use the mapping function γ in order to identify the real criterion of alternative a that corresponds to the j^{th} criterion of model $i \in I \setminus \{N\}$.

Constraints (8) and (9) construct the intermediate evaluation of alternative $a \in A$ on each criterion required by the top model by summing the values of the corresponding s variables. This evaluation is used to model the partial concordance index on the j^{th} criterion of the top MR-Sort model between alternative a and the bottom (respectively top) profile of the category (denoted through $K(a)$) to which it should be assigned.

Constraints (12) and (13) fix $\omega_{a,j}^-$ to w_j^n if $\delta_{a,j}^- = 1$ and to 0 otherwise. Constraints (14) and (15) fix $\omega_{a,j}^+$ to w_j^n if $\delta_{a,j}^+ = 1$ and to 0 otherwise.

Finally, constraints (16) and (17) model the overall assignment of alternative $a \in A$ by forcing the sum of $\omega_{a,j}^-$ to be greater or equal to the majority threshold of model n , i.e. making alternative a outrank the lower profile of its assigned category, and by forcing the sum of $\omega_{a,j}^+$ to be strictly lower than this threshold, i.e. making alternative a not outrank the upper profile of its assigned category.

Since this mathematical program is used to infer a feasible HMR-Sort model, we denote it as MIP_{feas} .

4.3. Robustness and inconsistencies

Since multiple sets of parameters values of an HMR-Sort model may perfectly model the preference statements of a DM, we first consider adapting the previous mathematical model in order to provide more robust solutions.

Our intention is to propose an HMR-Sort model such that slight variations in its parameters would not invalidate some of the expressed preference statements. We therefore seek to maximize the support of an alternative being assigned to a given category, which corresponds to maximizing the sum of the weights of the criteria on which it outranks the lower profile of the considered category and minimizing the sum of the weights of the criteria on which it outranks the upper profile. We do this for all MR-Sort models within our HMR-Sort model and denote the mathematical model as MIP_{rob} .

The new model adds a single continuous variable σ . Constraints (6) and (7) are updated as follows :

$$\sum_{\substack{j \in J^i \\ g_{a,\gamma_j^i} \geq b_{h,j}^i}} w_j^i \geq \lambda^i + \sigma + s_{a,h}^i - 1 \quad \forall a \in A, \forall i \in I \setminus \{n\}, \forall h \in \{1, \dots, k^i - 1\} \quad (19)$$

$$\sum_{\substack{j \in J^i \\ g_{a,\gamma_j^i} \geq b_{h,j}^i}} w_j^i + \epsilon \leq \lambda^i - \sigma + s_{a,h}^i \cdot (1 + \epsilon) \quad \forall a \in A, \forall i \in I \setminus \{n\}, \forall h \in \{1, \dots, k^i - 1\} \quad (20)$$

while constraints (16) and (17) are updated as follows :

$$\sum_{j \in J^n} \omega_{a,j}^- \geq \lambda^n + \sigma \quad \forall a \in A \quad (21)$$

$$\sum_{j \in J^n} \omega_{a,j}^+ + \epsilon \leq \lambda^n - \sigma \quad \forall a \in A \quad (22)$$

Finally, an objective function seeking to maximize σ is also included.

While the previous two mathematical models assume that an HMR-Sort model consistent with all of the DMs statements can be found, in practice this may not always be the case.

For both of the previously proposed mathematical models, we therefore define two new variants that seek to maximize the number of fulfilled assignments. We require a new set of binary variables $t_a, \forall a \in A$, which, when equal to 1 will enforce the assignment of alternative a to the desired category $K(a)$ and when equal to 0 will relax the constraints enforcing such an assignment.

In the case of MIP_{feas} , constraints (16) and (17) become :

$$\sum_{j \in J^n} \omega_{a,j}^- \geq \lambda^n - (1 - t_a) \quad \forall a \in A \quad (23)$$

$$\sum_{j \in J^n} \omega_{a,j}^+ + \epsilon \leq \lambda^n + (1 - t_a) \quad \forall a \in A \quad (24)$$

Furthermore, the objective function seeks to maximize $\sum_{a \in A} t_a$. We denote this program as $\text{MIP}_{\text{max-assig}}$.

In the case of MIP_{rob} , constraints (16) and (17) become :

$$\sum_{j \in J^n} \omega_{a,j}^- \geq \lambda^n + \sigma - (1 - t_a) \quad \forall a \in A \quad (25)$$

$$\sum_{j \in J^n} \omega_{a,j}^+ + \epsilon \leq \lambda^n - \sigma + (1 - t_a) \quad \forall a \in A \quad (26)$$

The objective function becomes a sequence of two objectives, the first seeking to maximize the number of fulfilled assignments while the second seeking to find a robust HMR-Sort model. The first objective takes precedence over the second, which can be modeled as $\sum_{a \in A} t_a + \sigma$, since σ will never exceed a value of 0.5 due to constraint (3). We denote this program as $\text{MIP}_{\text{max-assig-rob}}$.

5. Empirical validation

In order to test the proposed model we have set up a series of empirical tests on a set of artificially generated benchmarks. The main goals of these tests were to analyze: a) the computation time b) the ability of the proposed approach to restore the original model c) its ability to handle noisy data.

5.1. Experiment design and implementation details

The experimental design is depicted in Fig. 4. We randomly construct an initial HMR-Sort model, denoted with Ω , where:

- the number of models n , the number of criteria for each model $m^i, \forall i \in \{1, \dots, n\}$, and the number of categories for each model $k^i, \forall i \in \{1, \dots, n\}$ are randomly generated;
- the majority thresholds for each model $\lambda^i \in \{1, \dots, n\}$ are randomly generated in the $(0.5, 1]$ interval;
- the criteria weights for each model $w^i, \forall i \in \{1, \dots, n\}$ are randomly generated using the method described in [38];

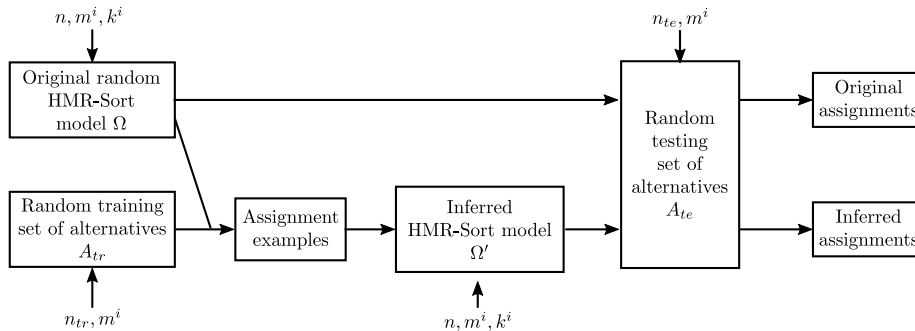


Figure 4: Design of experiments

- the limit profiles for each model $B^i, \forall i \in \{1, \dots, n\}$ are constructed by randomly generating $k^i - 1$ values in the $[0, 1]$ interval on each criterion and then sorting them in ascending order, so as to respect the dominance structure on the profiles;

A training set of n_{tr} alternatives A_{tr} is randomly generated, where the evaluations of the alternatives on each criterion are drawn from a uniform distribution in the unit interval. The assignment examples are constructed by applying the Ω model on these alternatives in order to extract their global assignments. The alternatives in A_{tr} together with their global assignments are then used by the proposed mathematical model in order to infer an HMR-Sort model, that we denote Ω' . This model is then compared to the original model (Ω), however, since multiple sets of parameters of an HMR-Sort model may express the same preferences of a DM, we do not employ a direct comparison of the parameters of these two models. Instead, we use a test set of n_{te} alternatives A_{te} generated in the same way as A_{tr} and compare their assignments by the two models. We motivate this firstly by the difficulty to propose a metric correctly combining the multiple and heterogeneous parameters of an HMR-Sort model. Secondly, our goal consists in modeling the preferences of a DM and not retrieving the precise values of the parameters of model which inherently seeks to approximate the DM's perspective.

We have constructed two sets of experiments based on the complexity of the generated HMR-Sort models. The first corresponds to smaller model instances where $n \in \{4, 5\}$, $m^i \in \{3, 4\}, \forall i \in \{1, \dots, n\}$ and $k_i \in \{3, 4\}, \forall i \in \{1, \dots, n\}$. The second corresponds to larger model instances where $n \in \{5, 6\}$, $m^i \in \{4, 5\}, \forall i \in \{1, \dots, n\}$ and $k_i \in \{4, 5\}, \forall i \in \{1, \dots, n\}$. In both cases $n_{te} = 5000$.

We have generated 50 HMR-Sort models (Ω) for each of the two sets of instances by randomly drawing the previously listed parameters from the given pools of values. The experiments have been performed using the solver IBM ILOG CPLEX 12.8 on an AMD Opteron TM 6176 SE machine with 250 GB RAM. However, for the smaller instances we allowed for a parallelism of 4 cores and up to 4 GB RAM for each execution, while for the larger instances we

used twice as much resources. In both cases, the results should be equivalent to using currently available personal computers. We have additionally set a 60 min timeout for each execution.

5.2. Results

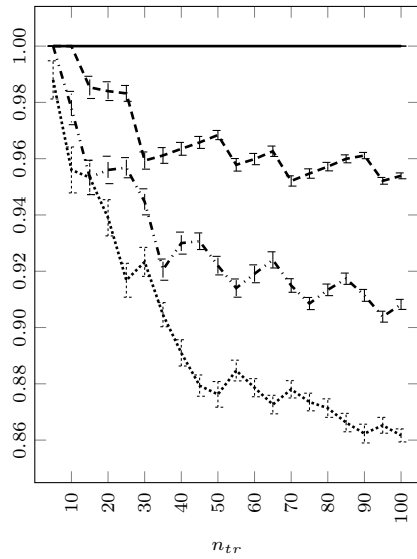
We begin by presenting the results of the MIP_{feas} and $MIP_{max-assign}$ approaches on the smaller sized instances in Figure 5. Each plot illustrates the evolution of a different indicator when sets of assignment examples of different sizes (from 5 to 100), are used as training data. Furthermore, four lines are depicted corresponding different degrees of errors, or noise, that are introduced in the assignments of the alternatives in the training data.

Figure 5a illustrates the average classification accuracy over the training data, or the capacity of the considered approach to retrieve a model that matches the assignments of the training set of alternatives. Error bars are also depicted in order to indicate the variability of these results. We observe that, when no errors are introduced in the assignment examples, the approach is able to completely restore them every time. However, as more and more alternatives receive errors in their assignments, the classification accuracy of the inferred model also decreases. We see how these values converge asymptotically to the 95%, 90% and 85% levels when 5%, 10% and 15% of the assignment examples get their categories changed respectively. When fewer assignment examples are considered, however, we are able to retrieve HMR-Sort models that satisfy a larger proportion of these assignment examples.

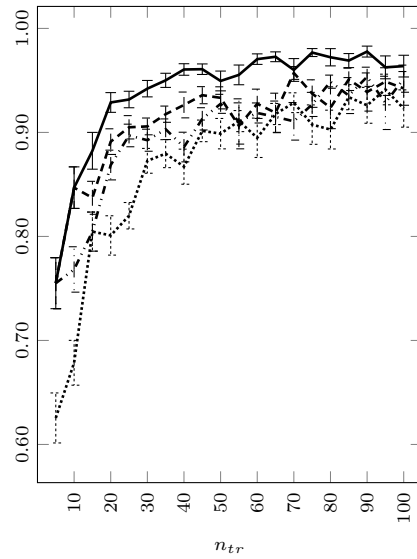
Figure 5b illustrates the average classification accuracy over the test data, or the capacity of the considered approach to retrieve the original model. We see that, as more assignment examples are considered, the inferred model comes closer and closer to the original model. Adding errors in the training data slightly decreases this ability, however to a smaller extent than when considering the classification accuracy over the training data.

Figure 5c shows the proportion of times the approach was forced to use $MIP_{max-assign}$ as MIP_{feas} was not able to find a model satisfying the entire set of assignment examples provided as training data. We again observe that when no errors are present in the data, there is no need to use $MIP_{max-assign}$. We also notice that when fewer assignment examples are provided, MIP_{feas} is able to find a model satisfying all assignments in the training data even when errors are included.

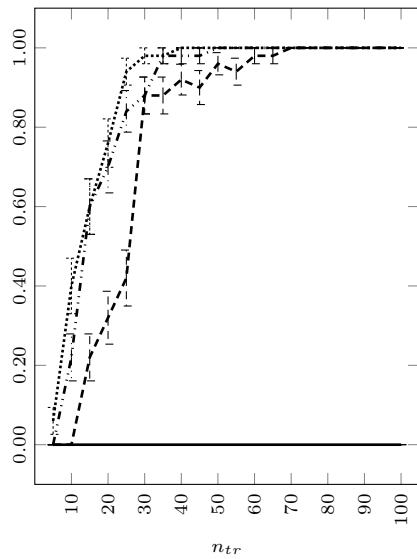
Finally, Figure 5d shows the average execution time of the entire approach, including the time needed to execute MIP_{feas} as well as the time needed for $MIP_{max-assign}$ when this program is needed. The plot illustrates the time axis on a logarithmic scale, which indicates an exponential increase in computation time as a function of the number of provided assignment examples. A slight top-off is observed which is due to the one hour limit that was enforced. We observe that the presence of errors in the training data significantly increases the time required to infer a model, which in part is due to the necessity of solving the more complex $MIP_{max-assign}$ in addition to MIP_{feas} . This is consistent with the fact that the largest increase in computation time is observed between



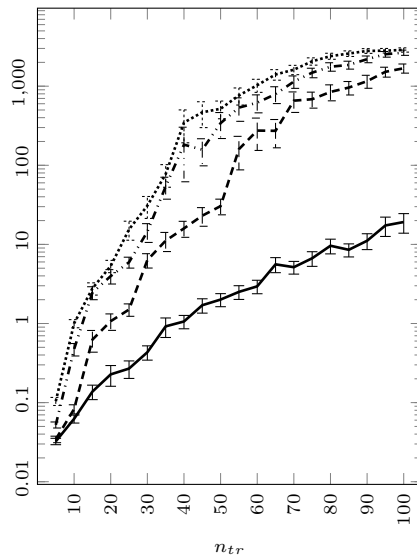
(a) Classification accuracy over train data



(b) Classification accuracy over test data



(c) Proportion of use of $MIP_{\max\text{-assig}}$



(d) Total execution time in seconds

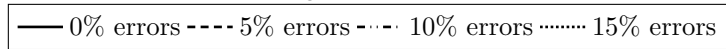


Figure 5: Results on the smaller instances using MIP_{feas} and $MIP_{\max\text{-assig}}$

the experiments containing 0% and 5% errors in the training data, while the differences between the experiments containing larger proportions of errors are less significant.

We continue by comparing these results to those when an objective function seeking to construct more robust HMR-Sort models is used. We present the results only for the cases where 0% and 10% errors are added to the training data in order to improve the readability of the results in Figure 6.

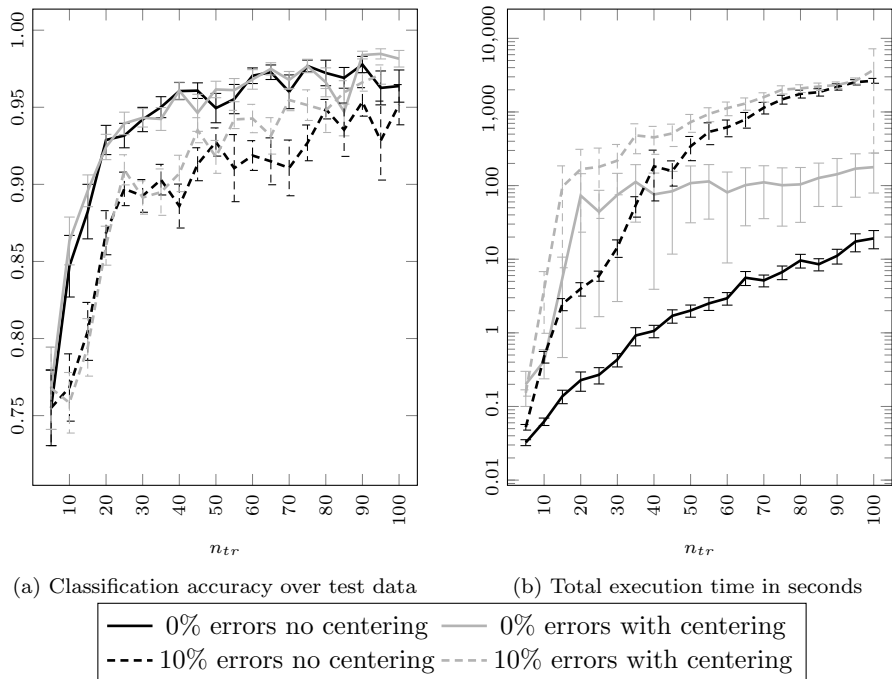


Figure 6: Results on the smaller instances comparing the approaches with and without a centering criterion

When no errors are present, we see that there are no significant differences w.r.t. the classification accuracy over the test data between the two approaches. Furthermore, the approach seeking to infer a robust HMR-Sort model takes a significant larger amount of time to finish. When errors are present in the training data, there is a slight increase in the classification accuracy of the second approach over the test data, however this comes at a significant cost w.r.t. the required computational time. We may therefore conclude that using the proposed robustness criterion in the mathematical program is not worth the added computational effort required.

Finally, we look at the results of the approaches not using a robustness criterion over the larger problem instances, which we depict in Figure 7.

We illustrate the results when only 0%, 5% and 10% errors are included in

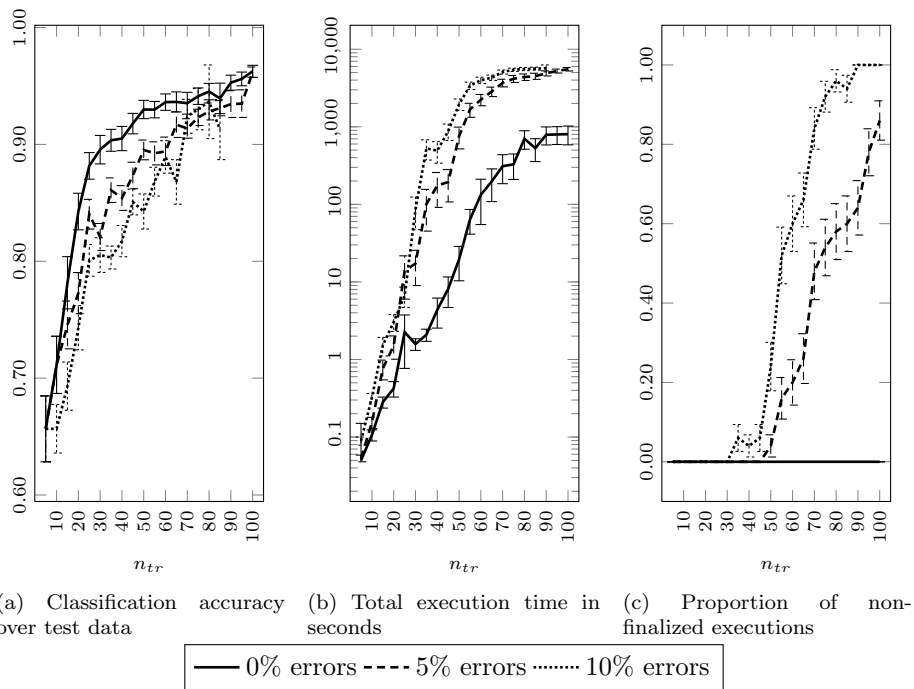


Figure 7: Results on the larger instances using MIP_{feas} and $MIP_{max-assig}$

the assignment examples due to the very high execution times required when introducing 15% errors. Figure 7a shows the classification accuracy over the test data and the results are consistent with those found on the smaller instances: increasing the number of assignment examples leads to more accurate inferred models, while errors in the assignment examples slightly lower this accuracy. Figure 7b shows the required execution times, and again, the results show an exponential increase with the number of assignment examples, capped by the one hour time limit that we imposed. Figure 7c shows the percentage of problem instances that could not be solved at all within this time limit. When no errors are present in the assignment example we can solve all instances with up to 100 assignment examples. However, when errors are present, starting from 50 assignment examples for 5% errors and from 30 assignment examples for 10% errors, we begin to not be able to solve some problem instances. For the case of 10% errors, we can also notice that no problem instance can be solved in one hour when considering more than 80 assignment examples.

6. Conclusions and perspectives

In this work we proposed an indirect inference approach based on mathematical programming in order to construct a hierarchical majority-rule sorting model that may be used to solve complex problems that can be divided into sub-problems and require the input of multiple DMs. Several tests have been performed on a large number of generated instances, which helped identify the limits for applying this approach on real problems. We find that, using currently available personal computers, we can infer accurate models within a relatively short amount of time for smaller two level hierarchical problems when no errors are present in the training data, while for larger problems we are limited to a smaller amount of input information when errors are present.

The proposed approach may easily be extended to other sorting approaches such as, for example, ELECTRE-Tri-B. Extending its applicability on larger problem instances and when using larger amounts of information in the form of assignment examples may be achieved by exploring metaheuristic alternatives to the exact inference approach. Furthermore, additional heterogeneous information related to the criteria weights and majority thresholds could also be included in order to boost the performance of the proposed approach. A further generalisation of the inference approach to more than two levels and to problem structures that are not limited to a tree structure in an additional perspective that merits further investigation.

References

- [1] B. Roy, *Multicriteria Methodology for Decision Aiding*, Kluwer Academic, Dordrecht, 1996.
- [2] A. Tsoukiàs, From decision theory to decision aiding methodology, *European Journal of Operational Research* 187 (1) (2008) 138–161.

- [3] S. Corrente, S. Greco, Słowiński, Multiple Criteria Hierarchy Process for ELECTRE Tri methods, *European Journal of Operational Research* 252 (1) (2016) 191–203. doi:10.1016/j.ejor.2015.12.053.
- [4] S. Corrente, S. Greco, Słowiński, Multiple Criteria Hierarchy Process with ELECTRE and PROMETHEE, *Omega* 41 (5) (2013) 820–846. doi:https://doi.org/10.1016/j.omega.2012.10.009.
- [5] A. Leroy, V. Mousseau, M. Pirlot, Learning the Parameters of a Multiple Criteria Sorting Method, in: R. Brafman, F. Roberts, A. Tsoukiàs (Eds.), *Algorithmic Decision Theory*, Vol. 6992, Springer, 2011, pp. 219–233.
- [6] R. Pelissari, A. José Abackerli, S. Ben Amor, M. Célia Oliveira, K. M. Infante, Multiple criteria hierarchy process for sorting problems under uncertainty applied to the evaluation of the operational maturity of research institutions, *Omega* 103 (102381) (2021). doi:10.1016/j.omega.2020.102381.
- [7] L. Del Vasto-Terrientes, A. Valls, Słowiński, ELECTRE-III-H: An outranking-based decision aiding method for hierarchically structured criteria, *Expert Systems with Applications* 42 (11) (2015) 4910–4926. doi:10.1016/j.eswa.2015.02.016.
- [8] S. Corrente, J. Figueira, S. Greco, Słowiński, A robust ranking method extending ELECTRE III to hierarchy of interacting criteria, imprecise weights and stochastic analysis, *Omega* 73 (2017) 1–17. doi:10.1016/j.omega.2016.11.008.
- [9] R. L. Keeney, H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, J. Wiley, New York, 1976.
- [10] Y. Zheng, S. Shekhar, A. A. Jose, S. K. Rai, Integrating context-awareness and multi-criteria decision making in educational learning, in: *Proceedings of the 34th ACM Symposium on Applied Computing*, 2019.
- [11] A. Frini, S. Ben Amor, MUPOM: A multi-criteria multi-period outranking method for decision-making in sustainable development context, *Environmental Impact Assessment Review* 76 (2019) 10 – 25.
- [12] L. Dias, J. Clímaco, ELECTRE TRI for groups with imprecise information on parameter values, *Group Decision and Negotiation* 9 (5) (2000) 355–377.
- [13] K. Jabeur, J.-M. Martel, An ordinal sorting method for group decision-making, *European Journal of Operational Research* 180 (3) (2007) 1272–1289. doi:10.1016/j.ejor.2006.05.032.
- [14] S. Greco, M. Kadziński, V. Mousseau, R. Słowiński, Robust ordinal regression for multiple criteria group decision: UTA^{GMS} -GROUP and $UTADIS^{GMS}$ -GROUP, *Decision Support Systems* 52 (3) (2012) 549–561. doi:10.1016/j.dss.2011.10.005.

- [15] M. Kadziński, S. Greco, R. Słowiński, Selection of a representative value function for robust ordinal regression in group decision making, *Group Decision and Negotiation* 22 (2013) 429–462. doi:10.1007/s10726-011-9277-z.
- [16] F. Lolli, A. Ishizaka, R. Gamberini, B. Rimini, M. Messori, FlowSort-GDSS – A novel group multi-criteria decision support system for sorting problems with application to FMEA, *Expert Systems with Applications* 42 (17) (2015) 6342–6349. doi:10.1016/j.eswa.2015.04.028.
- [17] P. A. Alvarez, I. A., L. Martínez, Multiple-criteria decision-making sorting methods: A survey, *Expert Systems with Applications* 183 (2021) 115368. doi:10.1016/j.eswa.2021.115368.
- [18] V. Mousseau, M. Pirlot, Preference elicitation and learning, *EURO Journal on Decision Processes* 3 (1-3) (2015) 1.
- [19] E. Jacquet-Lagrange, J. Siskos, Assessing a set of additive utility functions for multicriteria decision-making, the uta method, *European Journal of Operational Research* 10 (2) (1982) 151 – 164.
- [20] A. Ngo The, V. Mousseau, Using assignment examples to infer category limits for the ELECTRE TRI method, *Journal of Multi-Criteria Decision Analysis* 11 (1) (2002) 29–43.
- [21] I. Durbach, The use of the SMAA acceptability index in descriptive decision analysis, *European Journal of Operational Research* 196 (3) (2009) 1229 – 1237.
- [22] R. Lahdelma, J. Hokkanen, P. Salminen, SMAA - stochastic multiobjective acceptability analysis, *European Journal of Operational Research* 106 (1) (1998) 137 – 143.
- [23] K. Ciomek, M. Kadziński, T. Tervonen, Heuristics for selecting pair-wise elicitation questions in multiple criteria choice problems, *European Journal of Operational Research* 262 (2) (2017) 693 – 707.
- [24] N. Benabbou, P. Perny, P. Viappiani, Incremental elicitation of Choquet capacities for multicriteria choice, ranking and sorting problems, *Artificial Intelligence* 246 (2017) 152 – 180.
- [25] F. Lolli, E. Balugani, A. Ishizaka, R. Gamberini, M. A. Butturi, S. Marinello, B. Rimini, On the elicitation of criteria weights in PROMETHEE-based ranking methods for a mobile application, *Expert Systems with Applications* 120 (2019) 217–227. doi:10.1016/j.eswa.2018.11.030.
- [26] A. Khannoussi, A.-L. Olteanu, C. Labreuche, P. Meyer, Simple ranking method using reference profiles: incremental elicitation of the preference parameters, *4OR: A Quarterly Journal of Operations Research* (Jul. 2021). doi:10.1007/s10288-021-00487-w.

- [27] V. Mousseau, R. Słowiński, Inferring an ELECTRE TRI model from assignment examples, *Journal of Global Optimization* 12 (2) (1998) 157–174.
- [28] V. Mousseau, J. Figueira, J. Naux, Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results, *European Journal of Operational Research* 130 (2) (2001) 263–275.
- [29] L. Dias, J. Clímaco, On computing ELECTRE’s credibility indices under partial information, *Journal of Multi-Criteria Decision Analysis* 8 (2) (1999) 74–92.
- [30] L. Dias, V. Mousseau, J. Figueira, J. Clímaco, An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI, *European Journal of Operational Research* 138 (2) (2002) 332–348.
- [31] V. Mousseau, L. Dias, J. Figueira, C. Gomes, J. Clímaco, Resolving inconsistencies among constraints on the parameters of an MCDA model, *European Journal of Operational Research* 147 (1) (2003) 72–93.
- [32] V. Mousseau, L. Dias, J. Figueira, Dealing with inconsistent judgments in multiple criteria sorting models, *4OR* 4 (3) (2006) 145–158.
- [33] O. Sobrie, V. Mousseau, M. Pirlot, Learning a majority rule model from large sets of assignment examples, in: *ADT*, Vol. 8176 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 336–350.
- [34] A.-L. Olteanu, P. Meyer, Inferring the parameters of a majority rule sorting model with vetoes on large datasets, in: *DA2PL 2014: From Multicriteria Decision Aid to Preference Learning*, 2014, pp. 87–94.
- [35] O. Sobrie, V. Mousseau, M. Pirlot, A population-based algorithm for learning a majority rule sorting model with coalitional veto, in: H. Trautmann, G. Rudolph, K. Klamroth, O. Schütze, M. Wiecek, Y. Jin, C. Grimme (Eds.), *Evolutionary Multi-Criterion Optimization*, Springer International Publishing, Cham, 2017, pp. 575–589.
- [36] P. Meyer, A.-L. Olteanu, Handling imprecise and missing evaluations in multi-criteria majority-rule sorting, *Computers and Operations Research* 110 (2019) 135–147. doi:10.1016/j.cor.2019.05.027.
- [37] P. Meyer, A.-L. Olteanu, Integrating large positive and negative performance differences into multicriteria majority-rule sorting models, *Computers & Operations Research* 81 (2017) 216–230.
- [38] J. Butler, J. Jia, J. Dyer, Simulation techniques for the sensitivity analysis of multi-criteria decision models, *European Journal of Operational Research* 103 (3) (1997) 531–546.