



HAL
open science

Learning Non-Compensatory Sorting models using efficient SAT/MaxSAT formulations

Ali Tlili, Khaled Belahcene, Oumaima Khaled, Vincent Mousseau, Wassila Ouerdane

► **To cite this version:**

Ali Tlili, Khaled Belahcene, Oumaima Khaled, Vincent Mousseau, Wassila Ouerdane. Learning Non-Compensatory Sorting models using efficient SAT/MaxSAT formulations. *European Journal of Operational Research*, 2021, 298 (3), pp.979-1006. 10.1016/j.ejor.2021.08.017 . hal-03347526

HAL Id: hal-03347526

<https://hal.science/hal-03347526>

Submitted on 8 Jan 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

Learning Non-Compensatory Sorting models using efficient SAT/MaxSAT formulations

Ali Tlili*, Khaled Belahcène[†], Oumaima Khaled* , Vincent Mousseau*[‡],
Wassila Ouerdane*

May 31, 2021

Abstract

The Non-Compensatory Sorting model aims at assigning alternatives evaluated on multiple criteria to one of the predefined ordered categories. Computing parameters of the Non-Compensatory Sorting model compatible to a set of reference assignments is computationally demanding. To overcome this problem, two formulations based on Boolean satisfiability have recently been proposed to learn the parameters of the Non-Compensatory Sorting model from perfect preference information, i.e. when the set of reference assignments can be completely represented in the model. In this paper, two popular variants of the Non-Compensatory Sorting model are considered, the Non-Compensatory Sorting model with a unique profile and the Non-Compensatory Sorting model with a unique set of sufficient coalitions. For each variant, we start by extending the formulation based on a separation principle to the multiple category case. Moreover, we extend the two formulations to handle inconsistency in the preference information using the Maximum satisfiability problem language. A computational study is proposed to compare the efficiency of both formulations to learn the two Non-Compensatory Sorting models (with a unique profile and with a unique set of sufficient coalitions) from noiseless and noisy preference information.

Keywords— Multiple criteria analysis, Non-Compensatory Sorting, Preference Learning, SAT/MaxSAT

Introduction

Multiple Criteria Decision Analysis (MCDA) aims at developing decision-support models explicitly based on the construction of a set of criteria reflecting the relevant aspects of the decision-making problem. These n criteria ($\mathcal{N} = \{1, 2, \dots, n\}$ with $n \geq 2$) evaluate a set of alternatives $A = \{a, b, c, \dots\}$ under consideration with respect to different viewpoints. The MCDA literature considers different problem statements to formulate real-world decision problems; [45] distinguishes three problem statements: choice, sorting and ranking. As opposed to choice and ranking problem formulations which are comparative in nature, sorting formulates the decision problem in terms of the assignment of alternatives to one of the predefined ordered categories C^1, C^2, \dots, C^p , where C^1 (C^p , resp.) is the worst (the best, resp.) category. The assignment of an alternative to the appropriate category relies on its intrinsic value, and not on its comparison with other alternatives.

In this paper, we are interested in a specific sorting procedure: the Non-Compensatory Sorting (NCS) model [11, 12], which corresponds to a generalization and formal description of the Electre Tri procedure [21]. One of its

*MICS, CentraleSupélec, Université Paris-Saclay, Gif-Sur-Yvette, France

[†]Université de Technologie de Compiègne, CNRS, Heudiasyc, France

[‡]Corresponding author

specificity is to account for the alternative evaluations in an ordinal perspective avoiding compensation and enables to deal meaningfully with qualitative data.

We consider a decision aiding process in which two participants are involved to model a sorting problem using NCS: a decision maker (DM) looking for a recommendation and an analyst to support the DM in the search for this recommendation. Thus, the role of the analyst is to interact with the DM in order to help her to elaborate her preferences which are generally not fully predefined at the beginning of the decision process. The DM expresses preferences from which a specific NCS model is inferred. More specifically, the information supplied by the DM in order to specify the NCS sorting model are assignment examples (alternatives that should be assigned to a category). It should be highlighted that the construction of the learning set and the NCS model often results from a sequence of interactions between the DM and the analyst rather than in a one-step interaction.

In this perspective, the inverse Non-Compensatory Sorting problem (Inv-NCS, see section 3.1) takes as input a set of assignment examples, and computes (whenever it exists) an NCS sorting model which is consistent with this preference information. In other words, Inv-NCS learns the NCS parameters that perfectly match a set of desired outputs (assignment examples). Solving Inv-NCS problem is computationally difficult and have been proved to be NP-hard [6]. Mixed-integer linear formulations [35, 56] and heuristic resolution approaches [52, 51] have been proposed for Inv-NCS.

Recently, [5] proposed a SAT formulation of this problem which proves to be computationally more efficient than previous approaches. In this paper, we report a second SAT formulation for Inv-NCS [6] described in the context of two categories. We extend this second formulation to the multiple category case and perform numerical tests to compare the performance of these two SAT formulations. Nevertheless, in the SAT problem, it is assumed that the set of assignment examples is fully compatible with NCS. Therefore, we are interested in extending both SAT formulations to handle inconsistency in preference information which is often the case in real-world decision problems.

Indeed, in actual case studies, preference expressed by DMs are often inconsistent, due to the multiplicity of DMs, the fact that their preferences are not necessarily predefined and can evolve during the elicitation process. Handling inconsistencies when considering a set of preference statements on a set of multicriteria alternatives has been already tackled in the literature through mathematical programming (see, e.g. [41]) or the analysis of reciprocal preference relations (see, e.g. [26]). In this work, we consider handling inconsistency with MaxSAT language, where a SAT formulation is complemented with an implicit objective function, so that the number of satisfied clauses is maximal, allowing to best satisfy an unsatisfiable instance and consequently to best restore the assignment examples set.

The paper is organized as follows. In the first section, we propose an analysis of the recent literature on multicriteria sorting methods. Section 2 presents the NCS model. Inv-NCS, the problem of learning the parameters of NCS from assignment examples is defined in Section 3. In Section 4, we present the two SAT formulations for Inv-NCS. In Section 5, we extend SAT formulations with MaxSAT language, and Section 6 describes the empirical test design, the experimental results and a discussion. A final section groups conclusions and avenues for further research.

The main contributions of the paper concern the extension of the separation-based SAT formulation to the case of more than two categories (the end of section 4), the extension of the SAT formulations to MaxSAT to account for noisy input (Section 5), and empirical results providing insights on how our methods behave on actual data sets (Section 6). However, we also provide a comprehensive description of the NCS models (Section 2), so as to provide a self-contained text, as well as a brief survey of the recent literature concerning the elicitation of MCDA sorting models (Section 1).

1 Recent Literature on Multiple Criteria Sorting Methods

Many multicriteria sorting models have been proposed in the literature (see [17] for an overview). These multicriteria sorting models can be distinguished according to the way they model preferences: *(i)* the ones that model preferences using a multi-attribute value function (e.g., [16], [39], [33], [15], [47]), *(ii)* the ones that model preferences using outranking relations (e.g., [44], [43], [32],[20], [2]), and *(iii)* those which represent preferences using if-then rules (e.g., [24], [10], [31] [46]). Recently [30] proposed a method which corresponds to a hybridization of value-based and rule-based approaches.

Learning preference models from preference data to faithfully represent the DM's judgment has been considered since several decades in the literature. In the context of MCDA, a well-known example of such an approach is the UTA

method proposed in [27] in the case of an additive multicriteria value model. Learning an Electre Tri model (the initial multicriteria sorting procedure from which NCS was formalized, see [44]) from assignment examples was initially formulated using non-linear programming in [42].

A significant number of authors did focus on the robustness of sorting results. Some approaches are based on robust ordinal regression, e.g. [25], while some others focus on a stochastic approach, e.g. [54]. Another concern which emerged from the literature on sorting methods concerns the ability to explain the result to the decision maker in order to reinforce her trust (see [34], [4], [6]).

Since one decade, the literature on outranking-based sorting has widely expanded: based on the Electre Tri method [44], new sorting methods have been proposed defining categories using one or several limit profiles (see [19], [20]), one or several central profiles (see [1], [32], [2]). In parallel, several theoretical and axiomatic works have contributed to a better understanding of these methods (see [11, 12], [13]).

An additional significant advance in the literature concerns the new proposals in incremental elicitation of sorting methods (see [8, 9], [29]). These works propose a strategy to iteratively select questions to be asked to the decision maker in order to limit the number of questions.

Several new sorting methods allow to cope with possible interaction between criteria (see e.g. [36], [18]). The possibility to represent preferences in a hierarchical structure of criteria has also been considered (see [3]). Some authors also consider non monotone preferences in sorting methods (see [38], [40], [37]). These features enable sorting methods to account for more flexible preferences.

Another important trend in the sorting literature concerns the cross-fertilization between the field of MCDA sorting and preference learning ([23]). These two communities have now a common conference DA2PL (from Decision Analysis to Preference Learning) which takes place every second year since 2012. Among preference learning related work, one can cite [48], [38], [36], [37], [18], [46]. It should be noted that in this perspective, medical applications have been a fruitful application domain (see e.g [55], [49], [53]).

Coming back specifically to outranking-based sorting, previous works have proposed approaches to learn the parameters of an MR-Sort model (specific case of an NCS model in which the set of sufficient coalitions of criteria are defined using additive weights) based on a learning set. This MIP formulation minimizes the 0/1 loss, i.e. searches for a model that is compatible with as many examples as possible. Such MIP based exact approach has been extended to NCS 2-additive models ([51]). However, experimental results showed that such MIP approach becomes computationally prohibitive with a large number of assignment (learning an MR-Sort model with 100 alternatives, 5 criteria and 3 categories involve a MIP with 1100 binary variables, and the computing time exceeds 100 seconds).

To cope with the computational burden, a heuristic approach has been proposed to learn an MR-Sort model from assignment examples by [48, 52] which can handle large datasets, but losing optimality guaranty. More recently [5] defined a Boolean satisfiability formulation of Inv-NCS, which keeps optimality guarantee while enabling computations even for real-size datasets. In this paper, we continue and extend this work.

2 Non-Compensatory Sorting models

This section is devoted to the presentation of the Non-Compensatory Sorting model, introduced in [11, 12].

2.1 Basic notations

Multicriteria sorting aims at assigning alternatives to one of the predefined ordered categories $C^1 \prec \dots \prec C^p$. All alternatives in a set A are evaluated on n criteria, $\mathcal{N} = \{1, 2, \dots, n\}$; hence, an alternative $a \in A$ is characterized by its evaluation vector (a_1, \dots, a_n) , with $a_i \in \mathbb{X}_i$ denoting its evaluation on criterion i . Each criterion is equipped with a weak preference relation \succsim_i defined on \mathbb{X}_i . We assume, without loss of generality, that the preference on each criterion increases with the evaluation (the greater, the better). We denote by $\mathbb{X} = \prod_{i \in \mathcal{N}} \mathbb{X}_i$ the cartesian product of evaluation scales.

We recall the definitions of an upset and the upper closure of a subset w.r.t. a binary relation:

Definition 2.1. (*Upset and upper closure*). Let A be a set and \mathcal{R} a binary relation on A . An upset of (A, \mathcal{R}) is a

subset $\mathcal{B} \subseteq \mathcal{A}$ such that $\forall a \in \mathcal{A}, \forall b \in \mathcal{B}, a\mathcal{R}b \Rightarrow a \in \mathcal{B}$. The upper closure of a subset of $(\mathcal{A}, \mathcal{R})$ is the smallest upset of $(\mathcal{A}, \mathcal{R})$ containing it: $\forall \mathcal{B} \subseteq \mathcal{A}, cl_{\mathcal{A}}^{\mathcal{R}}(\mathcal{B}) := \{a \in \mathcal{A} : \exists b \in \mathcal{B} a\mathcal{R}b\}$

2.2 Sorting into two categories

In the Non-Compensatory Sorting model (NCS), the boundaries between categories are defined by limiting profiles. Therefore, a single profile corresponds to the case where alternatives are sorted between two ordered categories that we label as GOOD and BAD. A pair of parameters describe a specific sorting procedure:

- a limiting profile $b \equiv \langle b_i \rangle_{i \in \mathcal{N}}$ that defines, according to each criterion $i \in \mathcal{N}$, an upper set $\mathcal{A}_i \subset \mathbb{X}_i$ of approved values at least as good as b_i (and, by contrast, a lower set $\mathbb{X} \setminus \mathcal{A}_i \subset \mathbb{X}_i$ of disapproved values strictly worse than b_i), and
- a set \mathcal{T} of sufficient coalitions of criteria, which satisfies monotonicity with respect to inclusion.

These notions are combined into the following assignment rule:

$$\forall x \in \mathbb{X}, \quad x \in \text{GOOD} \iff \{i \in \mathcal{N} : x_i \succsim_i b_i\} \in \mathcal{T}$$

An alternative is considered as GOOD if, and only if, it is better than the limiting profile b according to a sufficient coalition of criteria.

2.3 Sorting into multiple categories

With p categories, the parameter space is extended accordingly, with approved sets $\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ defined by a set of limiting profiles $\langle b_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ and sufficient coalitions $\langle \mathcal{T}^k \rangle_{k \in [2..p]}$ declined per boundary.

The ordering of the categories $\{C^1 \prec \dots \prec C^p\}$ translates into a nesting of the sufficient coalitions: $\forall k \in [2..p]$, \mathcal{T}^k is an upset of $(2^{\mathcal{N}}, \subseteq)$ and $\mathcal{T}^2 \supseteq \dots \supseteq \mathcal{T}^p$, and also a nesting of the approved sets: $\forall i \in \mathcal{N}, \forall k \in [2..p]$, \mathcal{A}_i^k is an upset of $(\mathbb{X}_i, \succsim_i)$ and $\mathcal{A}_i^2 \supseteq \dots \supseteq \mathcal{A}_i^p$.

These tuples of parameters are augmented on both ends with trivial values: $\mathcal{T}^1 = \mathcal{P}(\mathcal{N})$, $\mathcal{T}^{p+1} = \emptyset$, and $\forall i \in \mathcal{N}$, $\mathcal{A}_i^1 = \mathbb{X}$, $\mathcal{A}_i^{p+1} = \emptyset$. With $\omega = (\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}, \langle \mathcal{T}^k \rangle_{k \in [2..p]})$, [12] defines the sorting function NCS_{ω} from \mathbb{X} to $\{C^1 \prec \dots \prec C^p\}$ with the *Non-Compensatory Sorting rule*:

$$NCS_{\omega}(x) = C^k \iff \begin{cases} \{i \in \mathcal{N} : x \in \mathcal{A}_i^k\} & \in \mathcal{T}^k \\ \text{and } \{i \in \mathcal{N} : x \in \mathcal{A}_i^{k+1}\} & \notin \mathcal{T}^{k+1} \end{cases} \quad (1)$$

Note that [11, 12] define a broader class of sorting method which includes vetoes which makes it possible for a single criterion to forbid the assignment to a class. Throughout this paper, we only consider NCS without veto; therefore, we should formally write NCS without veto all along with the paper. However, to facilitate the reading, we choose to write NCS even if we consider NCS models without a veto.

2.4 An illustrative example

A journalist prepares a car review for a forthcoming issue. She considers a number of popular car models and wants to sort them to present a sample of cars “selected for you by the editorial board” to the readers. This selection is based on four criteria: cost (€), acceleration (time, in seconds, to reach 100 km/h from full stop – lower is better), braking power and road holding, both measured on a qualitative scale ranging from 1 (lowest performance) to 4 (best performance). The performances of the six models are described in Table 1.

model	cost	acceleration	braking	road holding
m_1	16 973€	29.0 sec.	2.66	2.5
m_2	18 342€	30.7 sec.	2.33	3
m_3	15 335€	30.2 sec.	2	2.5
m_4	18 971€	28.0 sec.	2.33	2
m_5	17 537€	28.3 sec.	2.33	2.75
m_6	15 131€	29.7 sec.	1.66	1.75

Table 1: Performance table.

In order to assign these models to a category among C^{1*} (average) $\prec C^{2*}$ (good) $\prec C^{3*}$ (excellent), the journalist considers an NCS model:

- The attributes of each model are sorted between average (* / ■), good (** / ■) and excellent (***/ ■) by comparison to the profiles given in Table 2. The resulting labeling of the six alternatives according to each criterion is depicted in Figure 1 and Table 3.

profile	cost	acceleration	braking	road holding
b^{1*}	17 250€	30.0 sec.	2.2	1.9
b^{2*}	15 500€	28.8 sec.	2.5	2.6

Table 2: Limiting profiles.

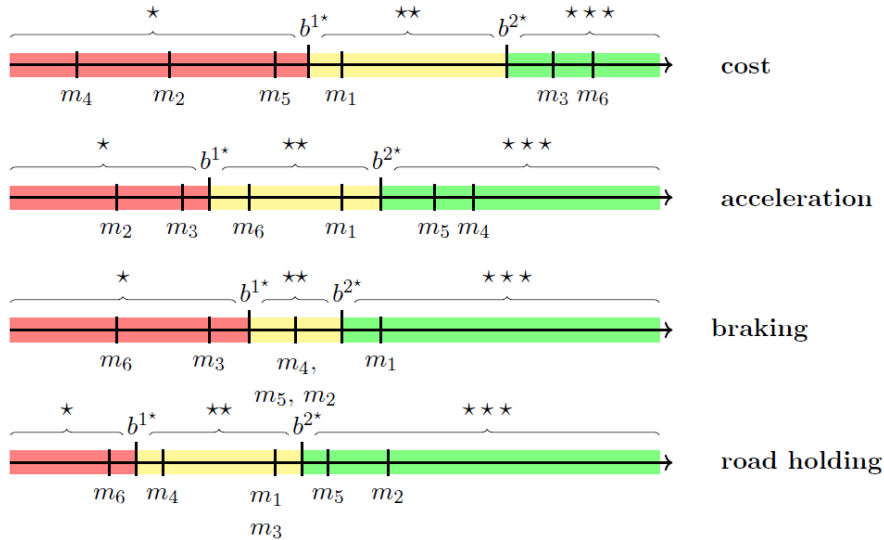


Figure 1: Representation of performances w.r.t. category limits.

- These appreciations are then aggregated by the following rule: *an alternative is categorized good or excellent if it is good or excellent on cost or acceleration, and good or excellent on braking or road holding. It is categorized excellent if it is excellent on cost or acceleration, and excellent on braking or road holding.* Being excellent on some criterion does not really help to be considered good overall, as expected from a Non-Compensatory model. Sufficient coalitions are represented on Figure 2. Finally, the model yields the assignments presented in Table 4.

model	cost	acceleration	braking	road holding
m_1	**	**	***	**
m_2	*	*	**	***
m_3	***	*	*	**
m_4	*	***	**	**
m_5	*	***	**	***
m_6	***	**	*	*

Table 3: Categorization of performances.

Alternatives	m_1	m_2	m_3	m_4	m_5	m_6
Assignment	**	*	**	**	***	*

Table 4: Alternative assignments.

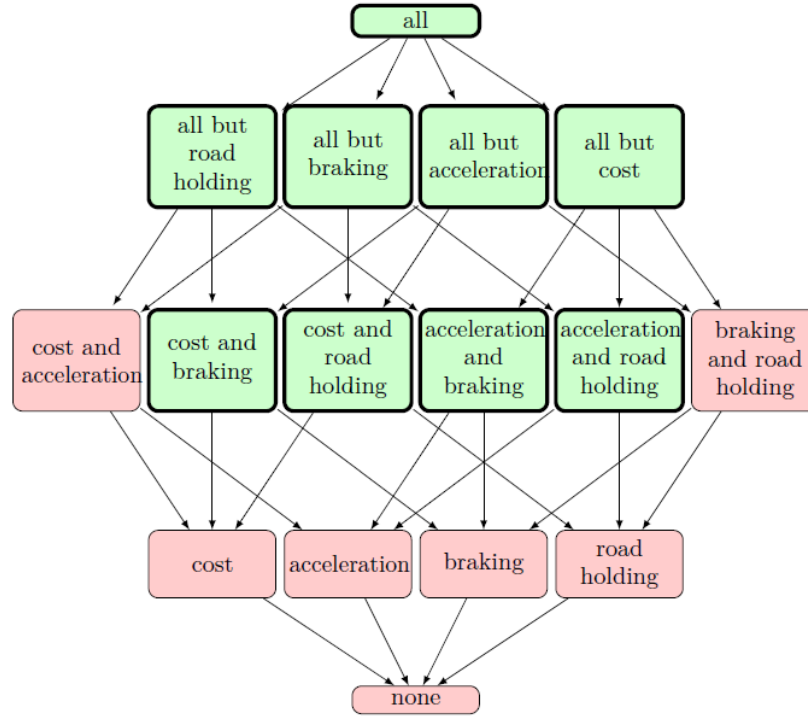


Figure 2: Sufficient (green/thick-bordered) and insufficient (red/thin-bordered) coalitions of criteria. Arrows denote coalition strength.

2.5 Variants of the NCS Model

In this section, we mention a number of variants of the Non-Compensatory Sorting model that can be found in the literature. Note that [11, 12] define the NCS class of sorting method, which includes the possibility of vetoes. In this paper, we only consider NCS without veto, but it should be highlighted that the broader class of NCS model can include vetoes, as depicted in Figure 3. Among NCS models without veto, there exist variants corresponding to simplifications of the model, with additional assumptions that restrict the parameters—limiting profiles and sufficient coalitions—either explicitly or implicitly.

The set of preference parameters – all the pairs $(\langle \mathcal{A} \rangle, \langle \mathcal{T} \rangle)$ can be considered too wide and too unwieldy for practical use in the context of a decision aiding process. Therefore, following [12], one may consider to explicitly restrict either the sequence of limiting profiles, or the sequence of sufficient coalitions:

- U^C -NCS: *Non-Compensatory Sorting with a unique set of sufficient coalitions*: $\mathcal{T}^2 = \dots = \mathcal{T}^p$;
- U^B -NCS: *Non-Compensatory Sorting with a unique limiting profile* $b^2 = \dots = b^p$ or, equivalently, $\forall i \in \mathcal{N}, \mathcal{A}_i^2 = \dots = \mathcal{A}_i^p$.

It worth to be noted that an NCS model which is in U^C -NCS and U^B -NCS simultaneously corresponds necessarily to a model with two categories (cf. the intersection colored in blue in Figure 3).

Another simplifying assumption consists in representing sufficient coalitions additively in an analogy to a voting setting: each criterion $i \in \mathcal{N}$ is assigned with a *voting power* $w_i \geq 0$ so that a given coalition of criteria $B \subseteq \mathcal{N}$ is deemed sufficient if, and only if, its combined voting power $\sum_{i \in B} w_i$ is greater than a given *qualification threshold* λ .

$$\exists \lambda, \langle w_i \rangle_{i \in \mathcal{N}} \in [0, 1] : \forall B \subseteq \mathcal{N}, B \in \mathcal{T} \iff \sum_{i \in B} w_i \geq \lambda. \quad (2)$$

With this rule, the sufficient coalitions are represented in a compact form which is more amenable to linear programming. This additive version of U^C -NCS is frequently called MR-Sort (for *majority rule sorting*) in the literature (see, e.g. [35]).

A more general way to describe possible interactions between criteria coalitions is to represent these coalitions using a capacity $\mu : 2^{\mathcal{N}} \mapsto [0, 1]$, with $\mu(\emptyset) = 0$, $\mu(\mathcal{N}) = 1$, and $\mu(B) \geq \mu(A)$, for all $A \subseteq B \subseteq \mathcal{N}$. The Möbius transform allows to express a capacity μ in another form: $\mu(A) = \sum_{B \subseteq A} m(B)$, $\forall A \subseteq \mathcal{N}$ with $m(B) = \sum_{C \subseteq B} (-1)^{|B|-|C|} \mu(C)$. The value $m(B)$ can be interpreted as the weight that is allocated to B as a whole. A capacity can be defined directly by its Möbius transform also called Möbius interaction. A Möbius interaction or Möbius mass m is a set function $m : 2^{\mathcal{N}} \mapsto [-1, 1]$ satisfying the hereafter conditions which guarantee that μ is monotone (see [14]):

$$\sum_{j \in K \subseteq J \cup \{j\}} m(K) \geq 0, \forall j \in \mathcal{N}, \forall J \subseteq \mathcal{N} \setminus \{j\} \text{ and } \sum_{K \subseteq \mathcal{N}} m(K) = 1.$$

Using such representation, it is possible to consider 2-additive (k-additive, resp.) capacities for which all the interactions involving more than 2 (k, resp.) criteria are equal to zero. 2-additive and k-additive MR-Sort (2-additive and k-additive U^C -NCS) are represented in Figure 3 (although not depicted, it is also possible to consider k-additive U^B -NCS).

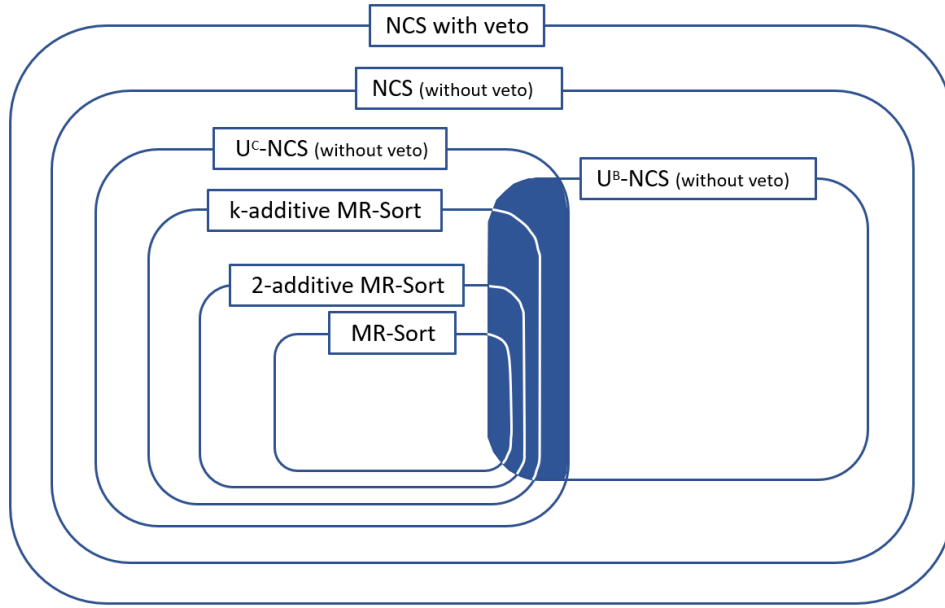


Figure 3: Variants of the NCS model.

3 Learning an NCS model from data

For a given decision situation, assuming the NCS model is relevant to structure the decision maker’s preferences, what should be the parameters’ values to fully specify the NCS model that corresponds to the decision-maker (DM) viewpoint? An option would be to simply ask the decision-maker to describe, to her best knowledge, the limit profiles between categories and to enumerate the minimal sufficient coalitions. To get this information as quickly and reliably as possible, an analyst could make good use of the *model-based elicitation strategy* described in [7], as it permits to obtain these parameters by asking the decision-maker to only provide holistic preference judgment – should some (fictitious) alternative be assigned to some category – and build the shortest questionnaire.

We opt for a more indirect setup, close to a machine learning paradigm [23], where a set of reference assignments is given and assumed to describe the decision-maker’s point of view, and the aim is to *extend* these assignments with an NCS model. In this context, we usually refer to an *assignment* as a function mapping a subset of *reference alternatives* $\mathbb{X}^* \subset \mathbb{X}$ to the ordered set of categories $C^1 \prec \dots \prec C^p$. These reference alternatives highlight values of interest on each criterion $i \in \mathcal{N}$, $\mathbb{X}_i^* = \bigcup_{x \in \mathbb{X}^*} \{x_i\}$. We refer to the problem of finding suitable preference parameters specifying a Non-Compensatory sorting model by Inv-NCS.

3.1 NCS and Inv-NCS

Instances. An *instance* of the Inv-NCS problem is a sextuple $(\mathcal{N}, \mathbb{X}, \langle \succsim_i \rangle_{i \in \mathcal{N}}, \mathbb{X}^*, \{C^1 \prec \dots \prec C^p\}, \alpha)$ where:

- \mathcal{N} is a set of criteria;
- \mathbb{X} is a set of *alternatives*;
- $\langle \succsim_i \rangle_{i \in \mathcal{N}} \in \mathbb{X}^2$ are *preferences* on criterion i , $i \in \mathcal{N}$, $\succsim_i \subset \mathbb{X}^2$ is a total pre-ordering of alternatives according to this criterion;
- $\mathbb{X}^* \subset \mathbb{X}$ is a finite set of *reference alternatives*;
- $\{C^1 \prec \dots \prec C^p\}$ is a finite set of *categories* totally ordered by *exigence*. We denote $C^{\succ k}$ (resp. $C^{\succ k}, C^{\preceq k}, C^{\prec k}$) the category interval $\{C^k \prec \dots \prec C^p\}$ (resp. $\{C^{k+1} \prec \dots \prec C^p\}, \{C^1 \prec \dots \prec C^k\}, \{C^1 \prec \dots \prec C^{k-1}\}$);

- $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$ is an *assignment* of the reference alternatives to the categories. Therefore, ‘ α^{-1} ’ is the associated inverse function i.e. for a given category C^h , $\alpha^{-1}(C^h) = \{x \in \mathbb{X}^* : x \in C^h\}$. For any comparison operator $\Delta \in \{\succ, \succeq, \prec, \preceq\}$, we also denote $\alpha^{-1}(C^{\Delta h}) := \{x \in \mathbb{X}^* : x \in C^h, C^k \Delta C^h\}$.

When referring to an instance, we often shorten this sextuple as ‘ α ’.

Parameters. Given a context, a *parameter* ω of the NCS model is a couple $(\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}, \langle \mathcal{T}^k \rangle_{k \in [2..p]})$, where the *sufficient coalitions* satisfy: $\forall k \in [2..p]$, \mathcal{T}^k is an upset of $(2^{\mathcal{N}}, \subseteq)$, and $\mathcal{T}^2 \supseteq \dots \supseteq \mathcal{T}^p$; and the *approved sets* satisfy $\forall i \in \mathcal{N}, \forall k \in [2..p]$, \mathcal{A}_i^k is an upset of $(\mathbb{X}_i, \preceq_i)$ and $\mathcal{A}_i^2 \supseteq \dots \supseteq \mathcal{A}_i^p$.

Sorting rule. Given a parameter $\omega = (\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}, \langle \mathcal{T}^k \rangle_{k \in [2..p]})$, augmented with trivial values $\mathcal{T}^1 := \mathcal{P}(\mathcal{N})$, $\mathcal{T}^{p+1} := \emptyset$, $\forall i \in \mathcal{N}$, $\mathcal{A}_i^2 = \mathbb{X}$, $\mathcal{A}_i^{p+1} = \emptyset$, NCS_ω is the function from \mathbb{X} to $\{C^1 \prec \dots \prec C^p\}$ satisfying:

$$NCS_\omega(x) = C^k \Leftrightarrow \begin{cases} \forall k' \leq k, & \{i \in \mathcal{N} : x \in \mathcal{A}_i^{k'}\} \in \mathcal{T}^{k'} \text{ and} \\ \forall k' > k, & \{i \in \mathcal{N} : x \in \mathcal{A}_i^{k'}\} \notin \mathcal{T}^{k'}. \end{cases} \quad (3)$$

This rule can be equivalently written as follows:

$$NCS_\omega(x) \in C^{\succeq k} \Leftrightarrow \{i \in \mathcal{N} : x \in \mathcal{A}_i^k\} \in \mathcal{T}^k. \quad (4)$$

Solutions. Given a context, a *solution* of the instance α of the Inv-NCS problem is a parameter ω of the NCS model such that $\forall x \in \mathbb{X}^*$, $\alpha(x) = NCS_\omega(x)$.

3.2 Literature related to Inv-NCS

Learning preference models from preference data to faithfully represent the DM judgment has been considered since several decades in the literature. In the context of MCDA, a well-known example of such an approach is the UTA method proposed in [27] in the case of an additive multicriteria value model. Learning an Electre Tri model (the initial multicriteria sorting procedure from which NCS was formalized, see [44]) from assignment examples was initially formulated using non-linear programming in [42]. A mixed-integer linear formulation was proposed by [35] to learn an additive majority rule sorting model (MR-Sort: additive NCS without veto) from a dataset; however, these approaches were not able to handle datasets corresponding to real-world problems. Recently, [28] proposed an enriched framework to elicit and Electre Tri B model and analyze its results.

To cope with the computational burden, a heuristic approach has been proposed to learn an MR-Sort model from assignment examples by [48, 52] which can handle large datasets, but losing optimality guaranty. More recently [5] defined a Boolean satisfiability formulation of Inv-NCS, which keeps optimality guarantee while enabling computations even for real-size datasets. In this paper, we continue and extend this work.

4 Boolean Satisfiability formulations for the Inv-NCS problem

This section is devoted to the presentation of two formulations of the inverse Non-Compensatory Sorting problem, first described respectively in [5] and [6], in the framework of Boolean satisfiability. They reduce the problem of finding the parameters of an NCS model faithfully reproducing a given assignment of alternatives to categories to the SAT problem of finding an assignment of Boolean variables that verify a given propositional formula written in conjunctive normal form¹.

¹For the convenience of EJOR readers, who might be more accustomed to the formalism of Mathematical Programming, we treat SAT as the tiny subset of MP where the variables are restricted to the $\{0, 1\}$ domain, the objective function is null, and the constraints are limited to linear forms of the type $\sum_{i \in C_j^+} x_i + \sum_{i \in C_j^-} (1 - x_i) \geq 1$, corresponding to the clause $\bigvee_{i \in C_j^+} x_i \vee \bigvee_{i \in C_j^-} \neg x_i$.

The two formulas stem from different representation strategies. One, detailed in section 4.1 and introduced in [5], establishes a bijection between the parameter space of the NCS model and the valuations of the propositional variables, and therefore introduces a number of variables that is exponential in the number of criteria. The other is detailed in section 4.3 and was introduced in [6]. It leverages a powerful representation theorem, detailed in section 4.2, that allows keeping implicit the set of coalitions, by introducing the notion of *pairwise separation*, formulated using pairs of alternatives given in the assignment.

Appendix A complements this section by providing previously unpublished formulations for the case where there are more than two categories, including the variants with a unique profile or a unique set of sufficient coalitions described in Section 2.5.

4.1 A SAT formulation for Inv-NCS based on coalitions

This section describes and extends a SAT formulation for Inv-NCS initially given in [5]. We provide here an informal presentation of the approach; formal justification can be found in [5]. The formulation Φ_α^C yielded by the encoding presented in this section is based on an explicit representation of the parameter space of the Non-Compensatory Sorting model—each the pairs are composed of a sequence of approved sets and a sequence of sufficient coalitions.

The explicit representation requires involving two families of binary variables. The first family defines the approved sets according to the set of criteria such that for a given alternative, level and criterion, the associated variable equals 1 if and only if the alternative is approved at the considered level according to the considered criterion. The second family of binary variables uniquely specifies the set of sufficient coalitions for each level i.e. given a coalition of criteria, the associated variable equals 1 if and only if the coalition is sufficient. The SAT formulation based on coalitions aims at learning both NCS parameters ($\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$, $\langle \mathcal{T}^k \rangle_{k \in [2..p]}$) from a set of assignment examples, thus, two types of clauses are considered. The first type of clauses defines these parameters and reproduces the structural conditions i.e.: the monotony of scales, approved sets and sufficient coalitions sets are ordered by inclusion and for each level the corresponding set sufficient coalitions is monotone by inclusion. The second type of clauses ensures the restoration of the assignment examples.

Variables. The Boolean function Φ_α^C operates on two types of variables:

- ‘*a*’ variables, indexed by a criterion $i \in \mathcal{N}$, an exigence level $k \in [2..p]$ and a reference value $x \in \mathbb{X}^*$, represent the approved sets \mathcal{A}_i^k , with the following semantic: $a_{i,k,x} = 1 \Leftrightarrow x \in \mathcal{A}_i^k$ i.e. x is approved at level k according to i ;
- ‘*t*’ variables, indexed by a coalition of criteria $B \subseteq \mathcal{N}$ and an exigence level $k \in [2..p]$, represent the sufficient coalitions \mathcal{T}^k , with the following semantic: $t_{B,k} = 1 \Leftrightarrow B \in \mathcal{T}^k$ i.e. the coalition B is sufficient at level k ;

Clauses. For a boolean function written in conjunctive normal form, the clauses are *constraints* that must be satisfied simultaneously by any antecedent of 1. The formulation Φ_α^C is built using six types of clauses:

- Clauses ϕ_α^{C1} ensure that each approved set \mathcal{A}_i^k is an upset of $(\mathbb{X}^*, \succsim_i)$: if for a criterion i and an exigence value k , the value x is approved, then any value $x' \succsim_i x$ must also be approved.
- Clauses ϕ_α^{C2} ensure that approved sets are ordered by a set inclusion according to their exigence level: if an alternative x is approved at exigence level k according to the criterion i , it should also be approved at exigence level $k' < k$.
- Clauses ϕ_α^{C3} ensure that each set of sufficient coalitions \mathcal{T} is an upset for inclusion: if a coalition B is deemed sufficient at exigence level k , then a stronger coalition $B' \supset B$ should also be deemed sufficient at this level.
- Clauses ϕ_α^{C4} ensure that a set of sufficient coalitions are ordered by inclusion according to their exigence level: if a coalition B is deemed insufficient at exigence level k , it should also be at any level $k' > k$.

- Clauses ϕ_α^{C5} ensure that each alternative is not approved by a sufficient coalition of criteria at an exigence level above the one corresponding to its assigned category.
- Clauses ϕ_α^{C6} ensure that each alternative is approved by a sufficient coalition of criteria at an exigence level corresponding to its assignment.

Definition 4.1. Given an instance of Inv-NCS with an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$, the boolean function Φ_α^C with variables $\langle a_{i,k,x} \rangle_{i \in \mathcal{N}, k \in [2..p], x \in \mathbb{X}^*}$ and $\langle t_{B,k} \rangle_{B \subseteq \mathcal{N}, k \in [2..p]}$, is defined as the conjunction of clauses:

$$\Phi_\alpha^C = \phi_\alpha^{C1} \wedge \phi_\alpha^{C2} \wedge \phi_\alpha^{C3} \wedge \phi_\alpha^{C4} \wedge \phi_\alpha^{C5} \wedge \phi_\alpha^{C6}$$

$$\begin{aligned} \phi_\alpha^{C1} &= \bigwedge_{i \in \mathcal{N}, k \in [2..p]} \bigwedge_{x' \succ_i x \in \mathbb{X}^*} (a_{i,k,x'} \vee \neg a_{i,k,x}) \\ \phi_\alpha^{C2} &= \bigwedge_{i \in \mathcal{N}, k < k' \in [2..p], x \in \mathbb{X}^*} (a_{i,k,x} \vee \neg a_{i,k',x}) \\ \phi_\alpha^{C3} &= \bigwedge_{B \subset B' \subseteq \mathcal{N}, k \in [2..p]} (t_{B',k} \vee \neg t_{B,k}) \\ \phi_\alpha^{C4} &= \bigwedge_{B \subseteq \mathcal{N}, k < k' \in [2..p]} (t_{B,k} \vee \neg t_{B,k'}) \\ \phi_\alpha^{C5} &= \bigwedge_{B \subseteq \mathcal{N}, k \in [2..p]} \bigwedge_{x \in \alpha^{-1}(C^{k-1})} (\bigvee_{i \in B} \neg a_{i,k,x} \vee \neg t_{B,k}) \\ \phi_\alpha^{C6} &= \bigwedge_{B \subseteq \mathcal{N}, k \in [2..p]} \bigwedge_{x \in \alpha^{-1}(C^k)} (\bigvee_{i \in B} a_{i,k,x} \vee t_{\mathcal{N} \setminus B, k}) \end{aligned}$$

Written as such, clauses of Φ_α^C are highly redundant, possibly threatening computational efficiency². Instead, it is sufficient to consider clauses where ordered elements in a pair are adjacent to each other.

Model variants. As discussed in Section 2.5, the NCS model has many variants. Φ_α^C can easily be modified to account for two popular restrictions of the model:

- U^B -NCS (Unique profiles): Drop the index k concerning the exigence level for the ‘ a ’ variables, ignore the conjunction over exigence levels for clauses ϕ_α^{C1} , and ignore clauses ϕ_α^{C2} altogether;
- U^C -NCS (Unique set of sufficient coalitions): Drop the index k concerning the exigence level for the ‘ t ’ variables, ignore the conjunction over exigence levels for clauses ϕ_α^{C3} and ignore clauses, ϕ_α^{C4} altogether.

4.2 A characterization based on pairwise separation

4.2.1 The case of two categories

The problem of finding simultaneously the sets of accepted values of the criteria and the sets of sufficient coalitions has been considered computationally difficult from the onset. In this light, the assumption of an additive representation of sufficient coalitions with the *majority rule* can be considered as a convenient way to keep the search somewhat tractable³. Indeed, when the accepted values are known, finding the parameters (the voting power of each criterion and the qualification threshold) of a suitable majority rule becomes a mere linear program with continuous variables and can be solved in polynomial time. It is possible to represent the NCS model with two categories in the MAVT paradigm, using full-fledged $|\mathcal{N}|$ -ary capacities, but the corresponding linear program requires $2^{|\mathcal{N}|}$ variables. This approach is deceptively difficult, though, and we shall see that, from the viewpoint of Computer Theory, Inv-NCS is actually not more difficult than its restriction to additive coalitions. This result comes from a simple series of observations. In the following, we suppose given a set of reference alternatives \mathbb{X}^* , an assignment $\alpha : \mathbb{X}^* \rightarrow \{ \text{GOOD}, \text{BAD} \}$, and a tuple of accepted values $\langle \mathcal{A}_i \rangle \in \mathcal{P}(\mathbb{X})^{|\mathcal{N}|}$ such that, for each point of view $i \in \mathcal{N}$, \mathcal{A}_i is an upset of (\mathbb{X}, \succ_i) .

²Even though SAT solvers often perform better on reasonably overconstrained problems.

³This assumption might also have some relevance w.r.t. intelligibility and parsimony.

Observably sufficient and insufficient coalitions. Consider the sets of coalitions defined by

$$\mathcal{S}_{\langle \mathcal{A}_i \rangle}(\alpha) := cl_{\mathcal{P}(\mathcal{N})}^{\supseteq} \left(\bigcup_{g \in \alpha^{-1}(\text{GOOD})} \{i \in \mathcal{N} : g \in \mathcal{A}_i\} \right), \quad (6)$$

$$\mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha) := cl_{\mathcal{P}(\mathcal{N})}^{\subseteq} \left(\bigcup_{b \in \alpha^{-1}(\text{BAD})} \{i \in \mathcal{N} : b \in \mathcal{A}_i\} \right). \quad (7)$$

Any coalition in $\mathcal{S}_{\langle \mathcal{A}_i \rangle}(\alpha)$ is a superset of the set of criteria according to which some GOOD alternative is accepted, and should, therefore, be accepted. Thus, $\mathcal{S}_{\langle \mathcal{A}_i \rangle}(\alpha)$ is a *lower bound* of the set of sufficient coalitions for any solution of Inv-NCS. Conversely, any coalition in $\mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha)$ is a subset of the set of criteria according to which some BAD alternative is accepted, and should, therefore, be rejected. Thus, $\mathcal{P}(\mathcal{N}) \setminus \mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha)$ is an *upper bound* of the set of sufficient coalitions for any solution of Inv-NCS.

Characterization of solutions of Inv-NCS. The parameter $(\langle \mathcal{A}_i \rangle, \mathcal{T})$ is a solution of the instance α of Inv-NCS if and only if:

$$\mathcal{S}_{\langle \mathcal{A}_i \rangle}(\alpha) \subseteq \mathcal{T} \subseteq \mathcal{P}(\mathcal{N}) \setminus \mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha) \quad (8)$$

Remarkably, this equation allows to characterize the positive instances of Inv-NCS without referring to the set of sufficient coalitions of a solution, solely by checking if the sets $\mathcal{T}_{\langle \mathcal{A}_i \rangle}(\alpha)$ and $\mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha)$ are disjoint. This leads to the following elegant and efficient characterization, based on the notion of *pairwise separation*.

Theorem 4.1. *An assignment α of alternatives to categories can be represented in the Non-Compensatory Sorting model if, and only if, there is a tuple $\langle \mathcal{A}_i \rangle \in \mathcal{P}(\mathbb{X})^{|\mathcal{M}|}$ such that:*

1. (Upset): *for each point of view $i \in \mathcal{N}$, \mathcal{A}_i is an upset of (\mathbb{X}, \succsim_i) ; and*
2. (Pairwise separation): *for each pair of alternatives $(g, b) \in \alpha^{-1}(\text{GOOD}) \times \alpha^{-1}(\text{BAD})$, there is at least one point of view $i \in \mathcal{N}$ such that $g \in \mathcal{A}_i$ and $b \notin \mathcal{A}_i$.*

This theorem provides a polynomial certificate for the positive instances of the Inv-NCS problem, thus proving its membership of the NP complexity class as a corollary. Proofs of Theorem 4.1, and of the NP-hardness of Inv-NCS can be found in [6]. The extension of this characterization to any number of categories is straightforward and is presented in the following section and Appendix A.

4.2.2 The case of more than two categories

The case where there are $p > 2$ categories $\{C^1 \prec \dots \prec C^p\}$ requires a few adaptations of the formulation. It relies mostly on the fact that an NCS model with p categories is, informally, the combination of $p - 1$ NCS models with two categories whose parameters satisfy the nesting conditions on the sufficient coalitions of criteria and the accepted values.

Given an assignment α and an exigence level $k \in [2..p]$, we define the set of alternatives assigned to categories better than and including C^k denoted $C^{\succeq k}$ and the set of alternatives assigned to categories worse than C^k denoted $C^{\prec k}$ as:

$$C^{\succeq k} = \bigcup_{h \in [k..p]} C^h; \quad C^{\prec k} = \bigcup_{h \in [2..k-1]} C^h$$

We extend equations (6) and (7) so that, at a given exigence level k , observably sufficient coalitions account for “good” alternatives in $C^{\succeq k}$ and observably insufficient coalitions account for “bad” alternatives in $C^{\prec k}$.

Definition 4.2. *(Observed sufficient and insufficient coalitions given approved sets). Given an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$, approved sets $\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ such that \mathcal{A}_i^k is an upset of $(\mathbb{X}_i, \succsim_i)$ and $\mathcal{A}_i^2 \supseteq \dots \supseteq \mathcal{A}_i^p$, we note, for any exigence level $k \in [2..p]$:*

$$\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}(\alpha) = cl_{\mathcal{P}(\mathcal{N})}^{\supseteq} \left(\bigcup_{g \in \alpha^{-1}(C^{\succeq k})} \{i \in \mathcal{N} : g \in \mathcal{A}_i\} \right)$$

$$\mathcal{F}_{\langle \mathcal{A}_i^k \rangle}(\alpha) = cl_{\mathcal{P}(\mathcal{N})}^{\subseteq} \left(\bigcup_{b \in \alpha^{-1}(C^{\prec k})} \{i \in \mathcal{N} : b \in \mathcal{A}_i\} \right)$$

By construction: each set $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^2(\alpha)$ is an upset for inclusion; the sets $\langle \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \rangle$ are nested (i.e. $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^2(\alpha) \subseteq \dots \subseteq \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^p(\alpha)$); each set $\mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$ is a lower set for inclusion; and the sets $\langle \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \rangle$ are nested (i.e. $\mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^p(\alpha) \subseteq \dots \subseteq \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^2(\alpha)$). Additionally, having disjoint observed sufficient and insufficient coalitions at every exigence level, i.e. $\forall k \in [2..p] \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \cap \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) = \emptyset$ is a necessary and sufficient condition for the existence of nested sets of coalitions $\langle \mathcal{T}^k \rangle_{k \in [2..p]}$ such that $\forall k \in [2..p], \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \subseteq \mathcal{T}^k \subseteq \mathcal{P}(\mathcal{N}) \setminus \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$.

Theorem 4.2. (Pairwise formulation of the Non-Compensatory Sorting model). *An assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$ can be represented in the Non-Compensatory Sorting model if, and only if, there are tuples $\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ such that:*

1. (Upset): *for each criterion $i \in \mathcal{N}$ and for each exigence level $k \in [2..p]$, $\langle \mathcal{A}_i^k \rangle$ is an upset of $(\mathbb{X}_i, \lesssim_i)$; and*
2. (Nesting): *the approved sets are nested according to their exigence level, i.e. for each criterion $i \in \mathcal{N}$, $\mathcal{A}_i^2 \subseteq \dots \subseteq \mathcal{A}_i^p$ (according to a given point of view, an alternative approved at some exigence level k is also approved at any lower exigence level level); and*
3. (Pairwise separation): *for any two exigence levels $k \leq k'$, for each pair of alternatives $(g, b) \in \alpha^{-1}(C^{k'}) \times \alpha^{-1}(C^{k-1})$, there is at least one point of view $i \in \mathcal{N}$ such that $g \in \mathcal{A}_i^k$ and $b \notin \mathcal{A}_i^k$.*

Proof : $[(1, 2, 3) \Rightarrow (\text{NCS})]$. Given a set of approved sets $\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ such that for each exigence level $k \in [2..p]$, \mathcal{A}_i^k is an upset of $(\mathbb{X}_i, \lesssim_i)$ satisfying conditions 1, 2 and 3, we consider the sets of coalitions $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$ and $\mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$ for each exigence level $k \in [2..p]$. According to the the remark just above, α can be represented in the NCS model iff $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \cap \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) = \emptyset, \forall k \in [2..p]$. Suppose this intersection is not empty for a given $k \in [2..p]$, and let $B \in \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \cap \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$. By definition of $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$, there is an exigence level $h \in [k..p]$ and an alternative $g \in \alpha^{-1}(C^h)$ such that $\{i \in \mathcal{N} : g \in \mathcal{A}_i^h\} \subseteq B$. By definition of $\mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$, there is an exigence level $h' \in [2..k]$ and an alternative $b \in \alpha^{-1}(C^{h-1})$ such that $B \subseteq \{i \in \mathcal{N} : b \in \mathcal{A}_i^{h'}\}$. Consequently, there is no criterion $i \in \mathcal{N}$ according to which $g \in \mathcal{A}_i^h$ and $b \notin \mathcal{A}_i^{h'}$, contradicting condition 3. Hence, $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) \cap \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha) = \emptyset$.

$[\neg(1, 2, 3) \Rightarrow \neg(\text{NCS})]$. It is obvious that condition 1 and condition 2 are essential to learn an NCS model with nested satisfactory values (enforced by condition 2) and nested sufficient coalitions sets (by construction). Suppose now that condition 1 and condition 2 are satisfied and let $k \in [2..p]$, $k' \in [k..p]$ a pair of exigence levels and (g, b) a pair of alternatives such that $g \in \alpha^{-1}(C^{k'})$, $b \in \alpha^{-1}(C^{k-1})$ and $[(k, k'), (b, g)]$ falsifies condition 3 i.e. $\{i \in \mathcal{N} : g \in \mathcal{A}_i^k\} \subseteq \{i \in \mathcal{N} : b \in \mathcal{A}_i^{k'}\}$. As $g \in \alpha^{-1}(C^{k'})$, the coalition of criteria $\{i \in \mathcal{N} : g \in \mathcal{A}_i^k\}$ is observably sufficient at level k' . As $b \in \alpha^{-1}(C^{k-1})$, the coalition of criteria $\{i \in \mathcal{N} : b \in \mathcal{A}_i^k\}$ is observably insufficient at level k , and even more so at level $k' \geq k$. Hence the intersection $\mathcal{S}_{\langle \mathcal{A}_i^k \rangle}^{k'}(\alpha) \cap \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}^k(\alpha)$ is nonempty, and α cannot be represented in NCS. ■

4.3 A SAT formulation for Inv-NCS based on pairwise separation conditions

The Boolean satisfiability formulation for learning a NCS model presented in this section, denoted Φ_α^P , was initially described in [6] but only focusing on the case with two categories $C^1 \equiv \text{BAD} \prec C^2 \equiv \text{GOOD}$. We extend this formulation to the multiple categories case to learn NCS, U^B -NCS and U^C -NCS.

4.3.1 Learning NCS in the case of two categories

The SAT formulation based on pairwise separation initially given in [6] corresponds to the SAT encoding of both conditions of the Theorem 4.1. First condition which ensures the monotony of scales is represented by a single family

of clauses and operates on the same variables as the SAT formulation based on coalitions. In the second condition, additional binary variables are defined in order to represent the separation between the alternatives. A unique family of logical clauses represent the separation concept of the theorem and additional clauses and binary variables are required in order to express this representation in SAT language.

Encoding. Similarly to the formulation Φ_α^C described in Section 4.1, the formulation Φ_α^P operates on two types of variables:

- ‘ a ’ variables, representing the approved sets, with the exact same semantics as their counterpart in Φ_α^C , i.e.

$$a_{i,x} = \begin{cases} 1, & \text{if } x \in \mathcal{A}_i \text{ i.e. } x \text{ is approved according to } i; \\ 0, & \text{else.} \end{cases}$$

- auxiliary ‘ s ’ variables, indexed by a criterion $i \in \mathcal{N}$, an alternative g assigned to **GOOD** and an alternative b assigned to **BAD**, assessing if the alternative g is positively separated from b according to the criterion i , i.e.

$$s_{i,g,b} = \begin{cases} 1, & \text{if } g \in \mathcal{A}_i \text{ and } b \notin \mathcal{A}_i; \\ 0, & \text{else.} \end{cases}$$

Φ_α^P is the conjunction of four types of clauses: ϕ_α^{P1} ensuring each \mathcal{A}_i is an upset, ϕ_α^{P2} ensuring $[s_{i,g,b} = 1] \Rightarrow [g \in \mathcal{A}_i]$, ϕ_α^{P3} ensuring $[s_{i,g,b} = 1] \Rightarrow [b \notin \mathcal{A}_i]$, and ϕ_α^{P4} ensuring each pair (g, b) is positively separated according to at least one criterion.

Definition 4.3. Given an instance of *Inv-NCS* with two categories and an assignment $\alpha : \mathbb{X}^* \rightarrow \{ \mathbf{BAD} \prec \mathbf{GOOD} \}$, we define the Boolean function Φ_α^P with variables $\langle a_{i,x} \rangle_{i \in \mathcal{N}, x \in \mathbb{X}^*}$ and $\langle s_{i,g,b} \rangle_{i \in \mathcal{N}, g \in \alpha^{-1}(\mathbf{GOOD}), b \in \alpha^{-1}(\mathbf{BAD})}$, as the conjunction of clauses:

$$\phi_\alpha^P = \phi_\alpha^{P1} \wedge \phi_\alpha^{P2} \wedge \phi_\alpha^{P3} \wedge \phi_\alpha^{P4}$$

$$\begin{aligned} \phi_\alpha^{P1} &= \bigwedge_{i \in \mathcal{N}} \bigwedge_{x' \succ_i x \in \mathbb{X}^*} (a_{i,x'} \vee \neg a_{i,x}) \\ \phi_\alpha^{P2} &= \bigwedge_{i \in \mathcal{N}, g \in \alpha^{-1}(\mathbf{GOOD}), b \in \alpha^{-1}(\mathbf{BAD})} (\neg s_{i,g,b} \vee \neg a_{i,b}) \\ \phi_\alpha^{P3} &= \bigwedge_{i \in \mathcal{N}, g \in \alpha^{-1}(\mathbf{GOOD}), b \in \alpha^{-1}(\mathbf{BAD})} (\neg s_{i,g,b} \vee a_{i,g}) \\ \phi_\alpha^{P4} &= \bigwedge_{g \in \alpha^{-1}(\mathbf{GOOD}), b \in \alpha^{-1}(\mathbf{BAD})} (\bigvee_{i \in \mathcal{N}} s_{i,g,b}) \end{aligned}$$

The formulation is compact: $O(|\mathcal{N}| \cdot |\mathbb{X}|^2)$ variables, $O(|\mathcal{N}| \cdot |\mathbb{X}|^2)$ binary clauses and $O(|\mathbb{X}|^2) |\mathcal{N}|$ -ary clauses, whereas the number of ‘ t ’ variables in the first formulation increases exponentially with the number of criteria.

Should ϕ_α^P be satisfiable, the set \mathcal{T} of sufficient coalitions is not uniquely identified by the values of ‘ a ’ and ‘ s ’ variables of one of its models. Indeed, if $\langle a_{i,x} \rangle, \langle s_{i,g,b} \rangle$ is an antecedent of **1** by ϕ_α^P , then the parameter $\omega = (\langle \mathcal{A}_i \rangle, \mathcal{T})$ with accepted sets defined by $\mathcal{A}_i = \{x \in \mathbb{X} : a_{i,x} = 1\}$ and any upset \mathcal{T} of $(\mathcal{P}(\mathcal{N}), \subseteq)$ of sufficient coalitions containing the upset $\mathcal{S}_{\langle \mathcal{A}_i \rangle}(\alpha)$ and disjoint from the lower set $\mathcal{F}_{\langle \mathcal{A}_i \rangle}(\alpha)$ is a solution of this instance. Therefore, among the sets of sufficient coalitions compatible with the values of ‘ a ’ and ‘ s ’ variables, we can identify two specific ones, \mathcal{T}_{max} and \mathcal{T}_{min} . We will also denote by \mathcal{T}_{rand} , a randomly chosen compatible set of sufficient coalitions.

4.3.2 Learning NCS with more than two categories

When there are more than two categories, the sets of variables and clauses need to be extended in order to characterize the NCS model.

- ‘ a ’ variables are also indexed by an exigence level $k \in [2..p]$, i.e.

$$a_{i,k,x} = \begin{cases} 1, & \text{if } x \in \mathcal{A}_i^k \text{ i.e. } x \text{ is approved according to } i \text{ at exigence level } k; \\ 0, & \text{else.} \end{cases}$$

- ‘ s ’ variables are also indexed by a *pair* of exigence levels $(k, k') \in [2..p]^2, k \leq k'$, with $g \in \alpha^{-1}(C^{k'})$, $b \in \alpha^{-1}(C^{k-1})$, so that

$$s_{i,k,k',g,b} = \begin{cases} 1, & \text{if } g \in \mathcal{A}_i^{k'} \text{ and } b \notin \mathcal{A}_i^k; \\ 0, & \text{else.} \end{cases}$$

These additional indices do not refer to new variables, but allow to tie the s variables representing pairwise separation to the a variables representing acceptance at the proper exigence level.

As it was introduced in [6], in the second formulation we learn the nested approved sets $\langle \mathcal{A}_i^k \rangle_{i \in \mathcal{N}, k \in [2..p]}$ with which we identify the nested sets of sufficient coalitions $\langle \mathcal{S}_{\langle \mathcal{A}_i \rangle}^k(\alpha) \rangle$ and insufficient coalitions $\langle \mathcal{F}_{\langle \mathcal{A}_i \rangle}^k(\alpha) \rangle$. Approved sets are constrained so that the intersection between the sets of observably sufficient and insufficient coalitions is empty. Leveraging Theorem 4.2, this ensures that the reference assignments are fully restored.

Definition 4.4. *Given an instance of Inv-NCS with an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$, we define the Boolean function $\Phi_\alpha^{P'}$ with variables $\langle a_{i,k,x} \rangle_{i \in \mathcal{N}, k \in \{2..p\}, x \in \mathbb{X}^*}$ and $\langle s_{i,k,k',g,b} \rangle_{i \in \mathcal{N}, k \in \{2..p\}, k' \in \{k..p\}, g \in \alpha(C^{k'}), b \in \alpha(C^{k-1})}$, as the conjunction of clauses:*

$$\Phi_\alpha^{P'} = \phi_\alpha^{P'1} \wedge \phi_\alpha^{P'2} \wedge \phi_\alpha^{P'3} \wedge \phi_\alpha^{P'4} \wedge \phi_\alpha^{P'5}$$

$$\begin{aligned} \phi_\alpha^{P'1} &= \bigwedge_{i \in \mathcal{N}, k \in [2..p]} \bigwedge_{x' \succ_i x \in \mathbb{X}^*} (a_{i,k,x'} \vee \neg a_{i,k,x}) \\ \phi_\alpha^{P'2} &= \bigwedge_{i \in \mathcal{N}, k < k' \in [2..p], x \in \mathbb{X}^*} (a_{i,k,x} \vee \neg a_{i,k',x}) \\ \phi_\alpha^{P'3} &= \bigwedge_{i \in \mathcal{N}, k \in [2..p], k' \in [k..p]} \bigwedge_{g \in \alpha^{-1}(C^{k'}), b \in \alpha^{-1}(C^{k-1})} (\neg s_{i,k,k',g,b} \vee \neg a_{i,k,b}) \\ \phi_\alpha^{P'4} &= \bigwedge_{i \in \mathcal{N}, k \in [2..p], k' \in [k..p]} \bigwedge_{g \in \alpha^{-1}(C^{k'}), b \in \alpha^{-1}(C^{k-1})} (\neg s_{i,k,k',g,b} \vee a_{i,k',g}) \\ \phi_\alpha^{P'5} &= \bigwedge_{k \in [2..p], k' \in [k..p]} \bigwedge_{g \in \alpha^{-1}(C^{k'}), b \in \alpha^{-1}(C^{k-1})} (\bigvee_{i \in \mathcal{N}} s_{i,k,k',g,b}) \end{aligned}$$

The remarks made about an efficient implementation of Φ_α^C are still valid: many clauses are redundant in $\phi_\alpha^{P'1}$ and $\phi_\alpha^{P'2}$ and can safely be ignored. The remark concerning the non-uniqueness of \mathcal{T} in the case of two categories also applies for more than two categories to \mathcal{T}^k which are not uniquely defined by $\Phi_\alpha^{P'}$.

Corollary 4.1. *Given a context, an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$ can be represented in the Non-Compensatory Sorting model if, and only if $\Phi_{\alpha, \text{NCS}}^{P'}$ is satisfiable.*

A specific analysis of how to extend Definition 4.3 to more than two categories when learning a U^B -NCS or a U^B -NCS model is detailed in Appendix A.

5 MaxSAT relaxations for Inv-NCS

The previous section introduced mathematical and computational tools addressing the *decision* problem: can a given assignment be represented in the Non-Compensatory Sorting model (or one of its variants)? This set of tools has an important theoretical significance, and can also serve as a base for practical applications—see e.g. [6] for an

application in an *accountability* setting, where the representation theorem (Theorem 4.1) is leveraged to provide procedural regularity certificates with good properties in terms of computational hardness and privacy preservation, or jurisprudential explanations, should the outcome of the sorting process be contested. Nevertheless, this approach is not suited to the problem of learning a suitable NCS model from real data, because it does not tolerate the presence of noise in the data. There are numerous reasons for the input data not to reflect perfectly the model, e.g.: imperfections in the assessment of performance according to some point of view; mistaken assignment of an alternative to a category; or simply the oversimplification of reality represented by the model.

In this section, we address this issue by providing a relaxation of the decision formulations: instead of finding a NCS model restoring all examples of the learning set (or, probably, die trying), we try to find the model that restores the most. This approach is similar to the *empirical risk minimization* approach that is central in Machine Learning for supervised classification problems, using the 0-1 loss. While it is a common practice in ML to use a convex surrogate of the 0-1 loss to immensely speed up the learning process, we embrace the computationally much more demanding exact approach, because we believe the benefits are high in terms of accountability—we are absolutely sure no one can challenge the output model on the basis of a better restoration of the learning set—while the computational cost can be kept low enough—because the number of criteria and of learning examples are often low in typical applications, and because we propose a computationally efficient approach.

We formulate the relaxed *optimization* problem of finding the subset of learning examples (reference alternatives together with their assignment) correctly restored of maximum cardinality with a *soft constraint* approach, using the language of weighted MaxSAT. This framework, derived from the SAT framework, is based on a conjunction of clauses $\bigwedge c_i$ where each clause c_i is given a non-negative weight w_i , and maximizes the total weight of the satisfied clauses. In order to translate exactly our problem in this language, we leverage two basic techniques: we introduce switch variables ‘ z ’ allowing to precisely monitor the soft clauses we are ready to see violated, as opposed to hard clauses that remain mandatory; and we use big-stepped tuples of weights w_1, \dots, w_k with $w_1 \gg \dots \gg w_k$ allowing to specify lexicographically ordered goals in an additive framework.

5.1 A MaxSAT relaxation for Inv-NCS based on coalitions

This section elaborates on the SAT formulation introduced in Section 4.1. The MaxSAT extension of the formulation obtained when following a strategy based on the explicit representation of coalitions of criteria is based on the specification of the reference alternative to relax in order to remove conflicts in the clauses. For this purpose, we define the following additional binary variables:

- ‘ z ’ variables, indexed by an alternative x , represent the set of alternatives properly classified by the inferred model, with the following semantic: $z_x = 1 \Leftrightarrow \alpha^{-1}(x) = NCS_\omega(x)$ i.e. the alternative x is properly classified

These variables are introduced in some clauses to serve as switches:

- For any exigence level $k \in [2..p]$, let $B \subseteq \mathcal{N}$ a coalition of criteria, and x an alternative assigned to C^{k-1} by α . If $z_k = 1$ and $B \subseteq \{i \in \mathcal{N} : x \in \mathcal{A}_i^k\}$ then $t_{B,k} = 0$. This leads to the following conjunction of clauses:

$$\phi_\alpha^{\widetilde{C5}} = \bigwedge_{B \subseteq \mathcal{N}, k \in [2..p]} \bigwedge_{x \in \alpha^{-1}(C^{k-1})} (\bigvee_{i \in B} \neg a_{i,k,x} \vee \neg t_{B,k} \vee \neg z_x)$$

- For any exigence level $k \in [2..p]$, let $B \subseteq \mathcal{N}$ a coalition of criteria, and x an alternative assigned to C^k by α . If $z_k = 1$ and $B \subseteq \{i \in \mathcal{N} : x \in \mathcal{A}_i^k\}$ then $t_{\mathcal{N} \setminus B, k} = 0$. This leads to the following conjunction of clauses:

$$\phi_\alpha^{\widetilde{C6}} = \bigwedge_{B \subseteq \mathcal{N}, k \in [2..p]} \bigwedge_{x \in \alpha^{-1}(C^k)} (\bigvee_{i \in B} a_{i,k,x} \vee t_{\mathcal{N} \setminus B, k} \vee \neg z_x)$$

The objective in the MaxSAT formulation is to maximize the portion of alternatives properly classified, this is the subject of the following soft clause:

$$\phi_\alpha^{goal} = \bigwedge_{x \in X^*} z_x \tag{10}$$

The MaxSAT extension of the first formulation is the conjunction of the first four clauses of the SAT formulation given in definition 4.1 and clauses ϕ_α^{C5} , ϕ_α^{C6} and ϕ_α^{goal} .

Clauses composing the conjunctions ϕ_α^{C1} , ϕ_α^{C2} , ϕ_α^{C3} , ϕ_α^{C4} , ϕ_α^{C5} and ϕ_α^{C6} are hard, associated to the weight w_{max} , and we associate to ϕ_α^{goal} the weight w_1 such that $w_{max} > |\mathbb{X}^*| w_1$.

Model variants. Same modifications as in the SAT formulation are required to learn U^B -NCS and U^C -NCS models with noisy preference information:

- U^B -NCS (Unique profiles): Drop the index k concerning the exigence level for the ‘ a ’ variables, ignore the conjunction over exigence levels for clauses ϕ_α^{C1} , and ignore clauses ϕ_α^{C2} altogether;
- U^C -NCS (Unique set of sufficient coalitions): Drop the index k concerning the exigence level for the ‘ t ’ variables, ignore the conjunction over exigence levels for clauses ϕ_α^{C3} and ignore clauses ϕ_α^{C4} altogether.

5.2 A MaxSAT relaxation for Inv-NCS based on pairwise separation conditions

This section elaborates on the SAT formulation introduced in Section 4.3, following a representation strategy based on the pairwise separation of alternatives.

In the case of two categories, switch variables ‘ z ’ have the same indexation and semantics as in the previous section. They are introduced in the clauses representing the pairwise separation constraints:

$$\phi_\alpha^{\widetilde{P4}} = \bigwedge_{g \in \alpha^{-1}(\text{GOOD}), b \in \alpha^{-1}(\text{BAD})} (\bigvee_{i \in \mathcal{N}} s_{i,g,b} \vee \neg z_b \vee \neg z_g)$$

They also appear in the clause ϕ_α^{goal} (see Eq. 10) formulating our objective of restoring as many learning examples as we can.

The weighted MaxSAT relaxation of the SAT formulation obtained following the representation strategy based on pairwise separation of alternatives, in the case of two categories, is the conjunction of clauses $\phi_\alpha^{P1} \wedge \phi_\alpha^{P2} \wedge \phi_\alpha^{P3} \wedge \phi_\alpha^{\widetilde{P4}}$, where each clause is hard and receives the weight w_{max} , and of the clause ϕ_α^{goal} with weight w_1 such that $w_{max} > |\mathbb{X}^*| w_1$.

The generalizations of this MaxSAT formulation to the case with multiple categories, including adaptations geared towards learning U^B -NCS and U^C -NCS variants of the Non-Compensatory Sorting model, are provided in Appendix B.

6 Computational study

In this section, we present an empirical study that evaluates the intrinsic and comparative performances of the approaches presented in Section 4 and 5. There are eight of them, depicted on Fig. 4 and specified by three binary parameters:

- the Non-Compensatory Sorting model of preference sought, either with a *unique* boundary *profile* (subscript U^P), or with a *unique* set of sufficient *coalitions* (subscript U^C);
- the representation strategy adopted, based either on the explicit representation of the *coalitions* of criteria (superscript C) or on the *pairwise* separation of alternatives (superscript P); and
- the problem description, either *deciding* whether an instance can be represented in the model (\mathcal{D}) with a SAT solver, or *optimizing* the ability of the model to represent the assignment (\mathcal{O}) with a MaxSAT solver.

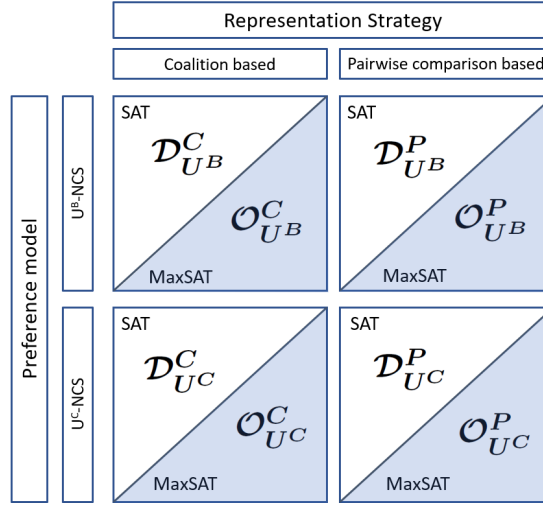


Figure 4: The eight approaches considered.

Note that the performances of \mathcal{D}_{UC}^C for learning U^C (Section 4.1) have already been proved to be superior to MIP approaches by [5].

6.1 Experimental design

The experimental plan consists of generating random instances of the Inv-NCS problem, applying one of the eight approaches described above, and measuring several performance indicators. We detail the instance generator, the implementation of the approaches and the indicators in the following sections.

6.1.1 Instance generation

Each instance consists of a set of alternatives \mathbb{X}^* (described by tuples of evaluations on a set of criteria \mathcal{N}), a set of categories $C^1 \prec \dots \prec C^p$, and the assignment of the former to the latter. We set the number of categories p to three. The set of alternatives is governed by two parameters –the number of criteria $|\mathcal{N}|$ and the number of reference alternatives $|\mathbb{X}^*|$ – that we consider exogenous and we try to assess their respective influence on the performance indicators. Note that this design is similar to a supervised classification context, where $|\mathbb{X}^*|$ and $|\mathcal{N}|$ are respectively the number of rows and columns of the dataset. Instances are sampled uniformly from the cube $[0, 1]^{|\mathcal{N}|}$: we have considered the least favourable case where all the criteria take continuous values.

The assignment of alternatives to categories depends on the type of model sought and the problem description. In order to ensure that preference data represents a real decision problem, we use a decision model to generate it, and, in particular, a model compatible to the Non-Compensatory stance we are postulating:

- In the case of U^C -NCS, we use an MR-Sort model for generating the learning set, a model that particularizes U^C by postulating the set of sufficient coalitions has an additive structure (see Section 2). It is randomly generated using the following procedure: a set of limit profiles $\langle b \rangle$ is generated by uniformly sampling $p - 1$ numbers in the interval $[0, 1]$ and sorting them in ascending order, for all criteria; the voting powers $\langle w \rangle$ are generated by sampling $|\mathcal{N}| - 1$ numbers in the interval $[0, 1]$, sorted and used as the cumulative sum of weights; the majority threshold λ is sampled with uniform probability in the interval $[0.5, 1]$.
- In the case of U^B -NCS we use a model with a unique profile and nested sets of sufficient coalitions of criteria at each exigence level, each with an additive structure, i.e., weights attached to criteria and a majority threshold. It is randomly generated using the following procedure: a single profile b is generated by uniformly sampling a tuple in $[0, 1]^{|\mathcal{N}|}$; the voting weights $\langle w \rangle$ are generated by sampling $|\mathcal{N}| - 1$ numbers in the interval $[0, 1]$, sorted and used as the cumulative sum of weights; the majority thresholds $\langle \lambda \rangle$ are then randomly chosen by sampling $p - 1$ numbers with uniform probability in the interval $[0.5, 1]$ and sorting them in ascending order.

Once the *ground truth* model is generated, which is by design compatible to the hypothesis class we are working with, we consider two ways of assigning alternatives to categories, depending on the problem formulation we are considering.

- For decision approaches, we directly assign the alternatives to categories according to the ground truth. Therefore, these approaches should always succeed in finding the parameters of a model extending the reference assignment.
- For optimization approaches, we introduce a proportion μ of assignment errors in the learning set. The assignment of a subset of reference alternatives is randomly replaced, with uniform probability, by the successor or predecessor category compared to the ground truth assignment.

6.1.2 Solving the instances

This experimental study is run on a laptop with Windows 10 (64 bit) equipped with an Intel(R) Xeon(R) CPU E5-1620 v4 @3.5GHz and 32 GB of RAM.

For decision approaches, we translate the assignment into a Boolean satisfaction problem, described by sets of variables and clauses, for both representation strategies and both preference models, as described in Section 4. The SAT instances are written in a file in DIMACS format, and passed to a command line SAT solver - CryptoMiniSat 5.0.1.

For optimization approaches, we translate the assignment into a Boolean satisfaction problem, described by sets of variables and clauses and an objective function, for both representation strategies and both preference models, as described in Section 5. The MaxSAT instances are passed to a command line MaxSAT solver QmaxSAT in the required format.

When using the representation strategy based on the explicit representation of the set of coalitions of criteria, each solution of the SAT/MaxSAT problem found by the solver can directly be interpreted in terms of parameters of an NCS model (either of the U^B or the U^C subtype). This is not exactly the case with the representation strategy based on pairwise separation of alternatives: the SAT/MaxSAT solution explicitly describes the approved sets of value on each criterion and at each exigence level (i.e. the boundary profiles), but the sets of sufficient coalitions are left implicit, and are solely described in terms of an upper and a lower bound. In the context of this experimental study, we are interested in resolute and precise decision models – hence it is necessary to complete this irresolute (or imprecise) strategy with a second strategy for picking a specific (nesting of) upset(s) of sufficient coalition inside the band of possible sets. We consider three such post-processing strategies: i) $\mathcal{T} = \mathcal{T}_{min}$, systematically returning the lower bound, ii) $\mathcal{T} = \mathcal{T}_{rand}$, returning a random nesting of upsets satisfying the constraints; and iii) $\mathcal{T} = \mathcal{T}_{max}$, returning systematically the upper bound.

6.1.3 Performance indicators

The performance indicators of interest are the computing time, the restoration rate (the proportion of the learning set correctly represented by the output model), and the generalization index measuring the alignment between the output model with the ground truth.

So as to monitor the learning process, we control the level of noise in the input data through the parameter μ , and we measure the proportion of reference assignments that are correctly restored by the learning process. This *restoration rate* should be equal to one in the case of approaches addressing the decision problem (as there is no noise), and at least equal to $1 - \mu$ for approaches addressing the optimization problem.

The computing performance is measured in practice, by solving actual instances of the problem and reporting the *computation time* required by the solver.

In order to appreciate how “close” a computed model to the ground truth from which the assignment examples were generated, and thus to monitor potential overfitting, we proceed as follows: we sample a large set of n profiles in $\mathbb{X} = [0, 1]^N$ and compute the assignment of these profiles according to the original and computed models. On this basis, we compute the *generalization index*: the proportion of “correct” assignments, i.e. profiles which are assigned to the same category by the ground truth and the inferred model.

6.2 Model retrieval with decision approaches

In this section, we study the behavior of the decision approaches, when fed with synthetic data matching the hypothesis (i.e. either coming from a specific U^B or U^C NCS model). More particularly, we monitor the restoration rate (which is expected to reach 100%), the computation time and the generalization index when applying each strategy (and, concerning the one based on the pairwise separation of alternatives, of three specific post-processing strategies concerning the choice of the nested sufficient coalitions), i.e. for the approaches \mathcal{D}_{UB}^C , \mathcal{D}_{UC}^C , \mathcal{D}_{UB}^P and \mathcal{D}_{UC}^P , as functions of the number of reference alternatives $|\mathbb{X}^*|$ and the number of criteria $|\mathcal{N}|$.

We explore a specific subset of the parameter space: we consider a baseline configuration, with 3 categories, 9 criteria and 128 reference alternatives, and we consider the configurations deviating from the baseline on a single parameter – either $|\mathbb{X}^*| = 128$ and $|\mathcal{N}| \in \{3, 5, 7, 11\}$, or $|\mathbb{X}^*| \in \{32, 64, 256, 512, 1024\}$ and $|\mathcal{N}| = 9$. For each configuration and for both models U^C and U^B , we sample 50 instances, then solve each of them according to both strategies.

6.2.1 Restoration rate

As expected, the restoration rate, for every model and strategy, is uniformly equal to one.

6.2.2 Computing time

For each NCS model (U^B and U^C), for each strategy under scrutiny (coalition based, and pairwise separation based), and for the set of considered parameters governing the input, the computation time ranges from below the tenth of a second to some dozens of minutes. Table 5 (respectively Table 6) depicts the distribution of the computation time for the baseline situation (128 reference assignments, 9 criteria and 3 categories) of implementing each strategy to learn a U^B model (resp. a U^C model). In this configuration, the strategy based on coalitions (\mathcal{D}^C) is slightly faster than the one based on pairwise separation (\mathcal{D}^P) when learning a U^B model and three times faster when learning a U^C model. The distribution of the computing time of each formulation is very tight around its center.

	\mathcal{D}_{UB}^C	\mathcal{D}_{UB}^P
Max	0.169s	0.293s
2 nd quartile	0.141s	0.184s
Median	0.126s	0.148s
1 st quartile	0.118s	0.111s
Min	0.108s	0.06s

Table 5: Computation time in the baseline configuration (128 ref. alternatives, 9 crit., 3 categ.) to learn a U^B model.

	\mathcal{D}_{UC}^C	\mathcal{D}_{UC}^P
Max	0.161s	0.584s
2 nd quartile	0.139s	0.389s
Median	0.131s	0.337s
1 st quartile	0.123s	0.256s
Min	0.104s	0.097s

Table 6: Computation time in the baseline configuration (128 ref. alternatives, 9 crit., 3 categ.) to learn a U^C model.

In order to assess the influence of the parameters governing the size and complexity of the input, we explore situations differing from the baseline on a single parameter:

- The number of reference assignments $|\mathbb{X}^*|$: Figures 5 and 6 indicate that the distribution of the computing time for both strategies and for both U^B and U^C models remains tightly grouped around its central value. It also shows that this value steadily increases with the number of reference assignments. For both strategies, the log-log plots are all consistent with a linear dependency between $\log t$ and $\log |\mathbb{X}^*|$, indicating the soundness of power law $t \propto |\mathbb{X}^*|^\beta$. The observed slopes are consistent with $\beta_C = 1$ (i.e. $t \propto |\mathbb{X}^*|$) for the representation strategy based on coalitions, and $\beta_P = 2$ (i.e. $t \propto |\mathbb{X}^*|^2$) for the representation strategy based on the pairwise separation of alternatives.

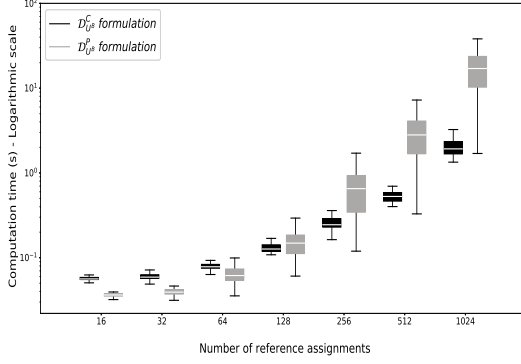


Figure 5: Computation time by number of ref. assignments (9 crit., 3 categ.) to learn a U^B model.

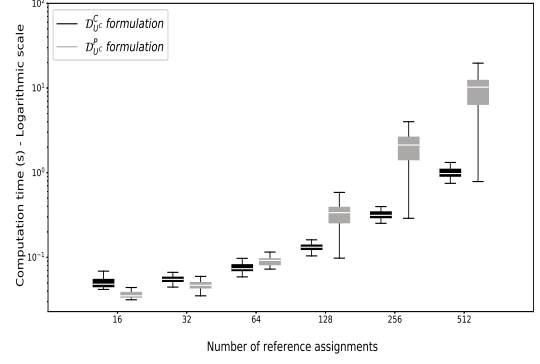


Figure 6: Computation time by number of ref. assignments (9 crit., 3 categ.) to learn a U^C model.

- The number of criteria $|\mathcal{N}|$: Figures 7 and 8 indicate for each NCS variants, the distribution of the computing time for both strategies. It can be observed that these series remain tightly grouped around their central value and this value steadily increases with the number of criteria. These observations are consistent with the hypotheses $t \propto |\mathcal{N}|$ for the representation strategy based on the pairwise separation of alternatives, and $t \propto 2^{|\mathcal{N}|}$ for the strategy based on coalitions of criteria.

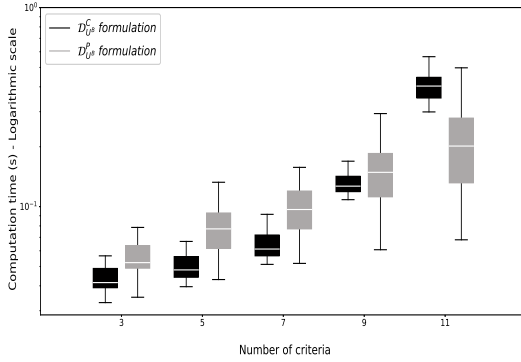


Figure 7: Computation time by number of criteria (128 ref. assignments and 3 categ.) to learn a U^B model.

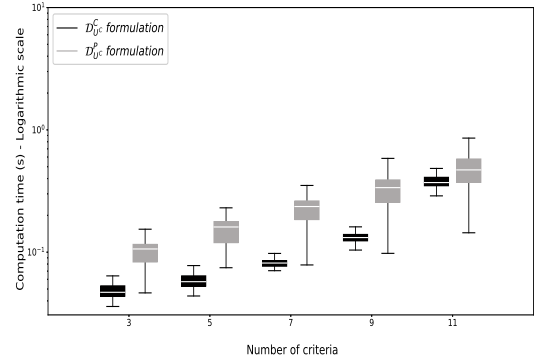


Figure 8: Computation time by number of criteria (128 ref. assignments and 3 categ.) to learn a U^C model.

6.2.3 Results on the ability of the inferred model to restore the original one

When applied to learn both NCS variants (U^B and U^C), the strategy based on pairwise separation returns an acceptable nesting of upset of sufficient coalitions, defined by lower and upper bounds. This strategy needs to be completed by a post-processing phase dedicated to pinpoint a single nesting of upsets. While this phase has no bearing on the restoration rate, and takes negligible time, it has a measurable impact on the generalization index.

To identify the upset that best restores the simulated sorting model ($1-U^B$ and MR-Sort), we study the three following post-processing strategies: $\mathcal{T} = \mathcal{T}_{min}$, $\mathcal{T} = \mathcal{T}_{rand}$ and $\mathcal{T} = \mathcal{T}_{max}$. T-Student tests ($\alpha = 5\%$) show that for U^B and U^C the generalization index when $\mathcal{T} = \mathcal{T}_{min}$ is always at least as good as the other two variants regardless the number of criteria, alternatives (and even categories for $p \in \{2, 3, 4, 5\}$); see for instance the baseline configuration Table 7. Consequently, for ease of presentation, we only plot results concerning the post-processing strategy $\mathcal{T} = \mathcal{T}_{min}$.

	\mathcal{D}_{UB}^C	\mathcal{D}_{UB}^P		
		$\mathcal{T} = \mathcal{T}_{min}$	$\mathcal{T} = \mathcal{T}_{rand}$	$\mathcal{T} = \mathcal{T}_{max}$
Max	96.4%	98%	97%	97%
2 nd quartile	92.3%	91.4%	89%	89%
Median	89%	90%	85.7%	85.7%
1 st quartile	83.4%	86.6%	80.8%	80.8%
Min	75.4%	79.8%	73%	73%

Table 7: Generalization index for both SAT formulations in the baseline configuration (128 reference assignments, 9 criteria and 3 categories) to learn a U^B model.

The first two columns of Table 7 depicts the distribution of the proportion of correct assignments (as compared to the ground truth) for the baseline situation (128 reference assignments, 9 criteria and 3 categories). T-Student test ($\alpha = 5\%$) shows that the difference between the two distributions is not significant.

Figures 9 and 10 represent the variations of the alignment of the models yielded by both formulations with the ground truth with respect to the problem settings when learning a U^B model (respectively U^C model) and applying each strategy. The experimental results display a tendency towards a degradation of this alignment as the number of criteria increases. Conversely, as expected, increasing the number of reference assignments noticeably enhances the generalization index, up to 100%. The implementations of both strategies seem to behave in a similar manner with respect to the variations of these parameters.

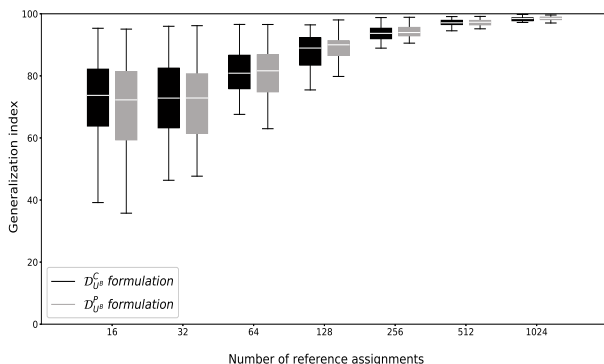


Figure 9: Generalization index by number of reference assignments (9 criteria and 3 categories) to learn a U^B model.

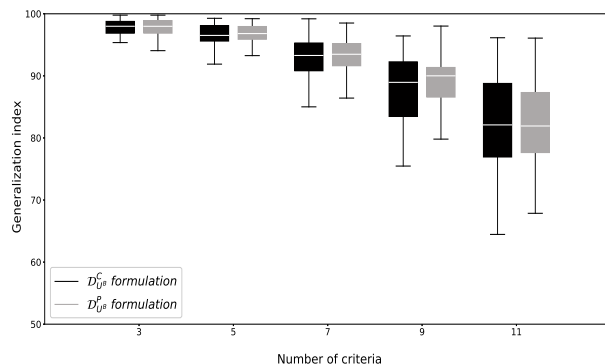


Figure 10: Generalization index by number of criteria (128 reference assignments and 3 categories) to learn a U^B model.

6.3 Tolerance for error with optimization approaches

In this section, we study the behavior of the optimization approaches, when fed with synthetic data that deviate from the model hypothesis (i.e. either coming from a specific U^B or U^C NCS model) in a controlled manner, through the incorporation of a proportion μ of noise. More particularly, we monitor the restoration rate (which is expected to reach at least $1 - \mu$), the computation time and the generalization index, when applying each strategy, i.e. for approaches \mathcal{O}_{UB}^C , \mathcal{O}_{UC}^C , \mathcal{O}_{UB}^P and \mathcal{O}_{UC}^P , as functions of the number of reference alternatives $|\mathbb{X}^*|$, the number of criteria $|\mathcal{N}|$, and the noise rate μ .

In this paper, the notion of noise on the learning set is defined as a misclassification of an alternative, i.e., an error from the decision maker in the choice of the category. More precisely, the assignment of a subset of reference alternatives is randomly replaced, with uniform probability, by the successor or predecessor category compared to the ground truth assignment. This is the way we have implemented the noise in our experiment ⁴.

⁴Note that there exist alternative ways to consider noisy expression of preferences. One of these is to consider that the errors in the learning set is related to the values/performances of alternatives in the learning set. Such noise is indeed relevant in applications

We explore a specific subset of the parameter space: we consider a baseline configuration, with 3 categories, 5 criteria, 128 reference alternatives and 10% noise rate, and we consider the configurations deviating from the baseline on a single parameter – $|\mathbb{X}^*| = 128$, $|\mathcal{N}| \in \{3, 7, 9, 11\}$ and $\mu = 0.1$; or $|\mathbb{X}^*| \in \{32, 64, 256\}$, $|\mathcal{N}| = 5$ and $\mu = 0.1$; or $|\mathbb{X}^*| = 128$, $|\mathcal{N}| = 5$ and $\mu \in \{.05, .15, .2\}$. For each configuration and for both models U^C and U^B , we sample 50 instances, then solve each of them according to both strategies.

6.3.1 Restoration rate

Plotting the restoration rate allows to monitor the learning process. The experimental results show that, when learning a given subtype of NCS model (either U^B or U^C), the models learned by implementing both strategies (either based on coalition or pairwise separation) reproduce the same portion of the learning set and at least $(1 - \mu) * |\mathbb{X}^*|$ assignment examples. This is some experimental evidence of the validity of the MaxSAT formulations stemming from both representation strategy.

The results display a tendency towards a degradation of the restoration rate distribution as the number of alternatives or the noise rate increases. Conversely, increasing the number of criteria noticeably enhances the restoration rate.

- The number of reference assignments $|\mathbb{X}^*|$: when the number of learning points (Figures 11 for U^B and 12 for U^C), we observe a convergence of the restoration rate towards its lower bound $(1 - \mu) * |\mathbb{X}^*|%$ (in this case 0.9): when the learning set is small, the computed model is flexible enough to reproduce almost all the learning set despite the errors; however, when the size of the learning set is large, as the computed model is more specific, the proportion of alternatives in the learning set whose assignment is not reproduced by the inferred model corresponds to the proportion of errors introduced in the learning set. Note however that alternatives in the learning set that are excluded when inferring the model do not necessarily correspond to the errors introduced in the learning set. However, the proportion of alternatives excluded when inferring the model is at most equal to the proportion of introduced errors. Also, it should be noted that the distribution of the restoration rate becomes more and more tightly grouped around its central value.

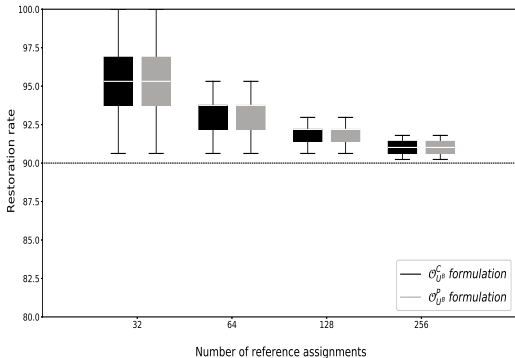


Figure 11: Restoration rate by number of reference assignments (5 criteria, 3 categories and 10% noise) to learn a U^B model.

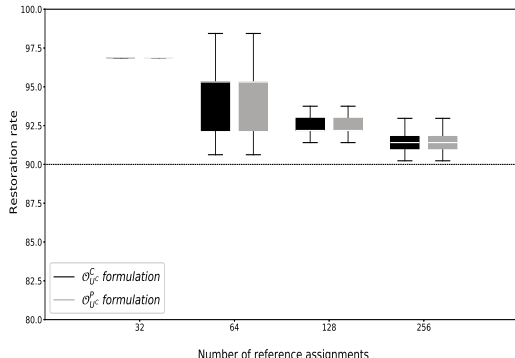


Figure 12: Restoration rate by number of reference assignments (5 criteria, 3 categories and 10% noise) to learn a U^C model.

- The number of criteria $|\mathcal{N}|$: Figures 13 (for U^B) and 14 (for U^C) show the variation of the restoration rate according to the number of criteria. Increasing the number of criteria makes the problem more flexible, and consequently noticeably enhances the restoration rate with a convergence towards 100%

where the learning set correspond to historical data in which performances of examples can be erroneous.

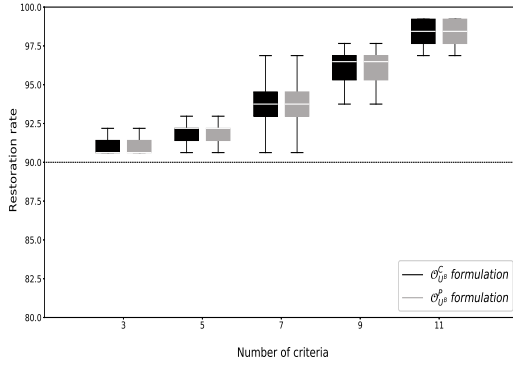


Figure 13: Restoration rate by number of criteria (128 reference assignments, 3 categories and 10% noise) to learn a U^B model.

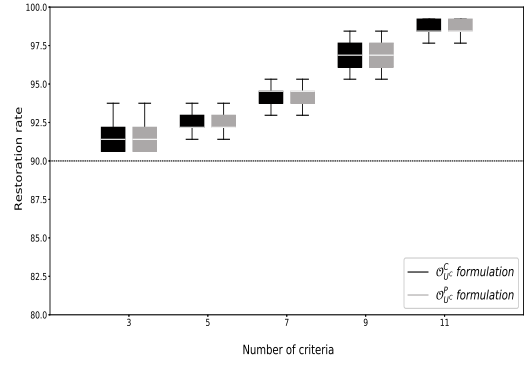


Figure 14: Restoration rate by number of criteria (128 reference assignments, 3 categories and 10% noise) to learn a U^C model.

- The noise rate μ : Figures 15 and 16 indicate that the restoration rate decreases linearly with the noise rate.

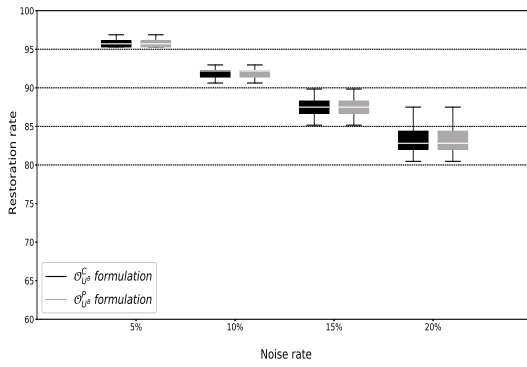


Figure 15: Restoration rate by noise rate (128 reference assignments, 5 criteria and 3 categories) to learn a U^B model.

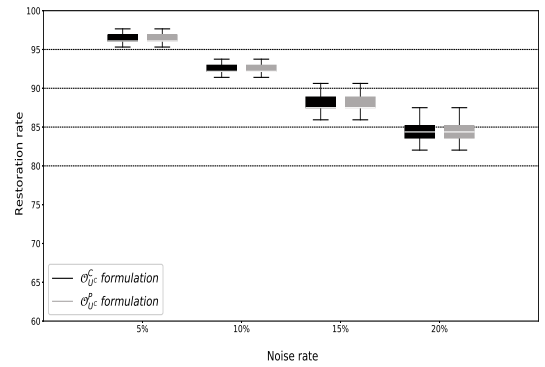


Figure 16: Restoration rate by noise rate (128 reference assignments, 5 criteria and 3 categories) to learn a U^C model.

6.3.2 Computing Time

Tables 8 and 9 show the distribution of the computation time in the baseline configuration (128 reference assignments, 5 criteria, 3 categories and 10 % noise) to learn both NCS models (U^B and U^C). When dealing with our baseline, applying the strategy based on the explicit representation of coalitions is 20 times faster than applying the strategy based on pairwise separation of alternatives, while this advantage was only threefold for the decision approaches (see e.g. Figures 7 and 8): the relaxation from SAT to MaxSAT seems to favor the strategy based on coalitions.

We investigate the influence of the parameters describing the instance.

	\mathcal{O}_{UB}^C	\mathcal{O}_{UB}^P
Max	0.337s	8.690s
2 nd quartile	0.272s	5.492s
Median	0.222s	4.132s
1 st quartile	0.176s	3.228s
Min	0.113s	1.195s

Table 8: Computation time to learn a U^B model in the baseline config. (128 ref. alt., 5 crit., 3 categ. and 10% noise).

	\mathcal{O}_{UC}^C	\mathcal{O}_{UC}^P
Max	0.996s	18.121s
2 nd quartile	0.554s	11.7s
Median	0.352s	8.161s
1 st quartile	0.242s	5.323s
Min	0.131s	1.582s

Table 9: Computation time to learn a U^C model in the baseline config.(128 ref. alt., 5 crit., 3 categ. and 10% noise).

- The number of reference assignments \mathbb{X}^* : Figures 17 and 18 indicate that the distribution of the computing time for the two MaxSAT-formulations and for both U^B and U^C models remains tightly grouped around its central value. It also shows that this value steadily increases with the number of reference assignments, consistently with the power laws found in Section 6.2.2, i.e. $t \propto |\mathbb{X}^*|$ for the representation strategy based on coalitions of criteria, and $t \propto |\mathbb{X}^*|^2$ for the representation strategy based on the pairwise separation of alternatives.

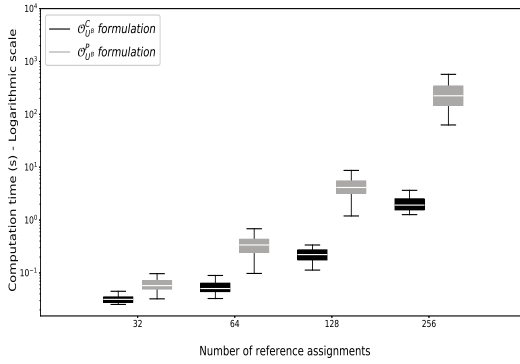


Figure 17: Computation time by number of reference assignments (5 criteria, 3 categories, 10% noise) to learn a U^B model.

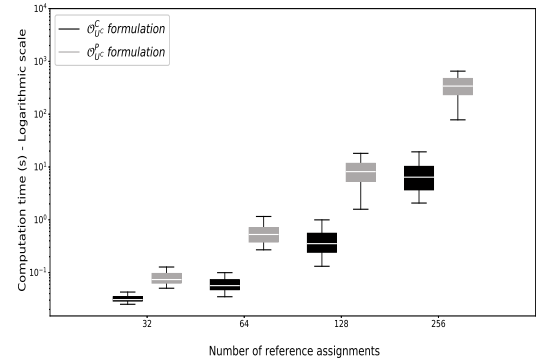


Figure 18: Computation time by number of reference assignments (5 criteria, 3 categories, 10% noise) to learn a U^C model.

- The number of criteria $|\mathcal{N}|$: Figures 19 and 20 indicates that the distribution of the computing time, when applying both strategies and learning both models U^C and U^B , remains tightly grouped around its central value. These results remain consistent to the models proposed in Section 6.2.2: $t \propto |\mathcal{N}|$ for the representation strategy based on the pairwise separation of alternatives, $t \propto 2^{|\mathcal{N}|}$ for the representation strategy based on coalitions of criteria.

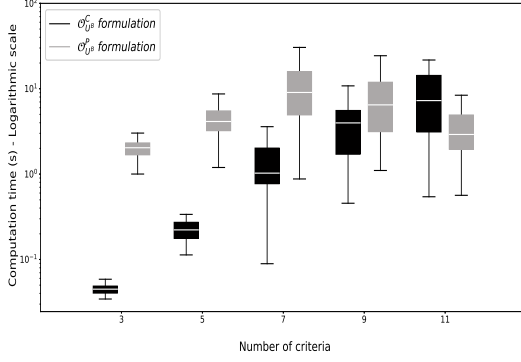


Figure 19: Computation time by number of criteria (128 reference assignments, 3 categories and 10% noise) to learn a U^B model.

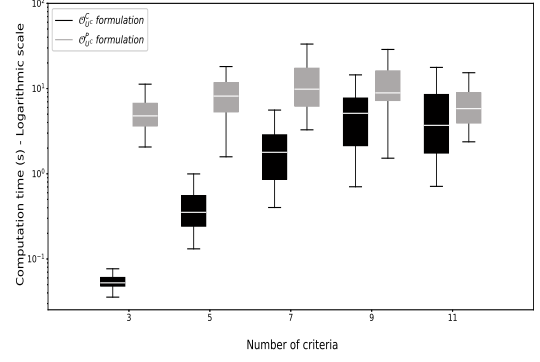


Figure 20: Computation time by number of criteria (128 reference assignments, 3 categories and 10% noise) to learn a U^C model.

- The noise rate μ : The distribution of the computation time for both MaxSAT formulations remains tightly grouped around its central value, and $\log t$ increases linearly (with a low slope) with the noise rate (Figures 21 and 22).

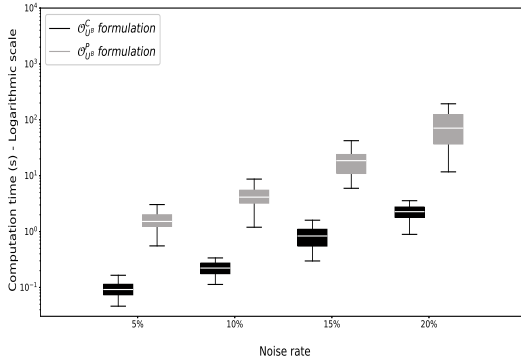


Figure 21: Computation time by noise rate (128 reference assignments, 5 criteria and 3 categories) to learn a U^B model.

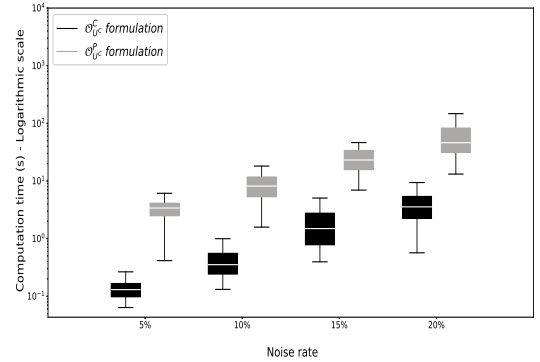


Figure 22: Computation time by noise rate (128 reference assignments, 5 criteria and 3 categories) to learn a U^C model.

6.3.3 Results on the ability of the inferred model to restore the original one

The observations made in Section 7, concerning the irresoluteness of the approaches implementing the representation strategy based on pairwise separation, remain valid when considering MaxSAT relaxations. Adopting the same notations as the SAT formulations, T-Student tests show that for both models U^B and U^C the generalization index when $\mathcal{T} = \mathcal{T}_{min}$ is always at least as good as the other two variants regardless of the number of criteria, alternatives, categories and the noise rate (see for instance the baseline configuration Table 10). The rule of thumb proposed in Section 7 remains valid when transposed to optimization approaches implemented via a MaxSAT solver – the post-processing strategy $\mathcal{T} = \mathcal{T}_{min}$ yields the best results, and is the only one represented on the subsequent figures.

	\mathcal{O}_{UB}^C	\mathcal{O}_{UB}^P		
		$\mathcal{T} = \mathcal{T}_{min}$	$\mathcal{T} = \mathcal{T}_{rand}$	$\mathcal{T} = \mathcal{T}_{max}$
Max	97.7%	99.2%	98.8%	98.8%
2 nd quartile	95.6%	96.5%	95.9%	95.9%
Median	94.4%	95.4%	94.6%	94.5%
1 st quartile	92.2%	93.5%	92.4%	92.6%
Min	87.7%	90%	88.2%	88%

Table 10: Generalization index in the baseline configuration (128 reference assignments, 5 criteria, 3 categories and 10% noise) when learning a U^B model, for both representation strategies and three post-processing strategies.

The first two columns of Table 10 depicts the distribution of the generalization index for both MaxSAT formulations for the baseline situation (128 reference assignments, 5 criteria, 3 categories and 10% noise) for learning a U^B model (respectively a U^C model). For both models, the two distributions are almost the same with a slight difference on the median.

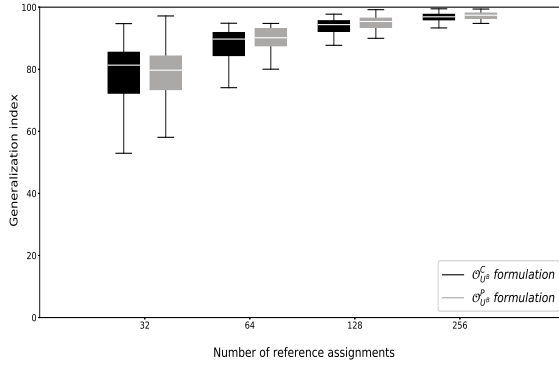


Figure 23: Generalization index by number of reference assignments (5 criteria, 3 categories and 10% noise) to learn a U^B model.

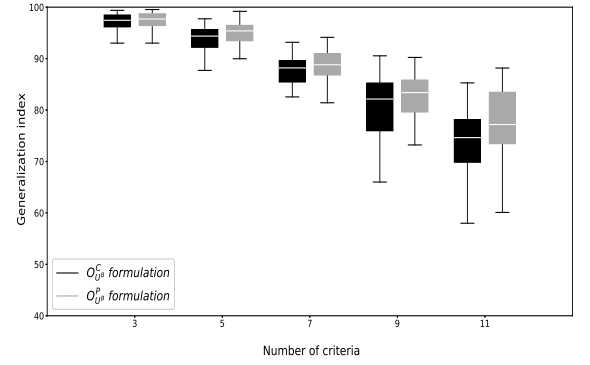


Figure 24: Generalization index by number of criteria (128 reference assignments, 3 categories and 10% noise) to learn a U^B model.

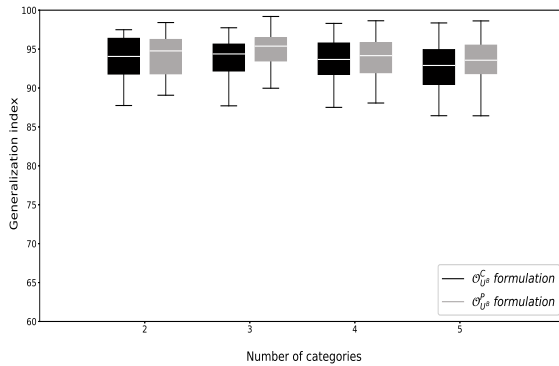


Figure 25: Generalization index by number of categories (128 reference assignments, 5 criteria and 10% noise) to learn a U^B model.

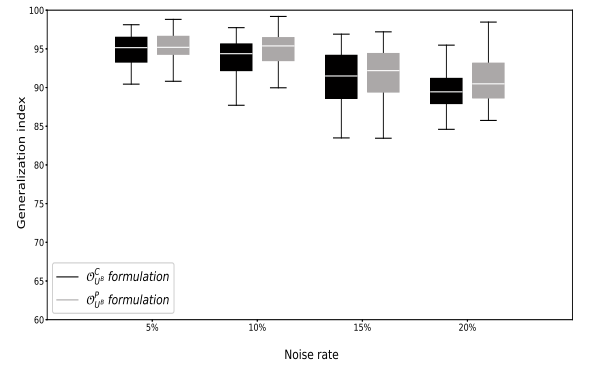


Figure 26: Generalization index by noise rate (128 reference assignments, 5 criteria and 3 categories) to learn a U^B model.

Figures 23, 24, 25 and 26 present the variations of the alignment of the computed U^B models (respectively U^C models) yielded by both MaxSAT formulations with the ground truth. For both NCS variants, the experimental results display a tendency towards a degradation of this alignment as the number of criteria or the number of categories increases. Conversely, as expected, increasing the number of reference assignments noticeably enhances the generalization index.

The two formulations seem to behave in a similar manner with respect to the modification of these parameters. And finally, the generalization rate decreases linearly with the noise rate.

6.4 Discussion

In this section, we discuss the influence of input parameters (number of criteria, and the size of the learning set) on the computing time, the ability to restore learning sets, and to generalize of both representation strategies (the one based on explicit representation of coalitions, and the one based on pairwise separation of alternatives). The discussion focuses on both problem descriptions: decision (SAT) and optimization (MaxSAT) for learning both variants of NCS (U^B and U^C). The results obtained provides (i) the empirical confirmation of results which were expected, and (ii) insights for an analyst who wishes to use the proposed learning algorithms in an decision-aiding case study.

6.4.1 Empirical confirmation of expected results

Computation time:

On the one hand, for each NCS variants (U^B and U^C) and for both SAT and MaxSAT problem descriptions, the number of reference assignments impacts linearly the computation time of the coalitions-based representation strategy, and quadratically the computation time of the pairwise separation representation). On the other hand, the coalitions-based representation strategy depends exponentially on the number of criteria, and this dependence remains linear for the separation-based representation.

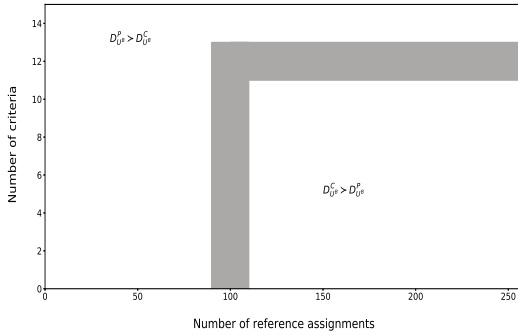


Figure 27: Computation time of SAT problems by number of reference assignments and number of criteria (3 categories) to learn a U^B model

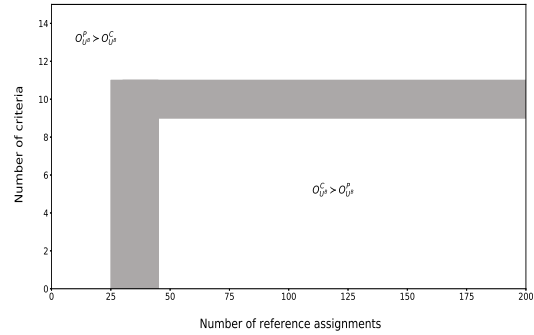


Figure 28: Computation time of MaxSAT problems by number of reference assignments and number of criteria (3 categories and 10% noise) to learn a U^B model

For a fixed number of criteria, when increasing the number of reference assignments, the coalition-based representation becomes faster than the separation-based representation (as the size of the learning set impacts the computing time linearly for the coalition-based representation, and quadratically for the separation-based representation).

Conversely, for a fixed number of reference assignments, when increasing the number criteria, the separation-based representation becomes faster than the coalition-based representation (as the number of criteria impacts the computing time exponentially for the coalition-based representation, and linearly for the separation-based representation).

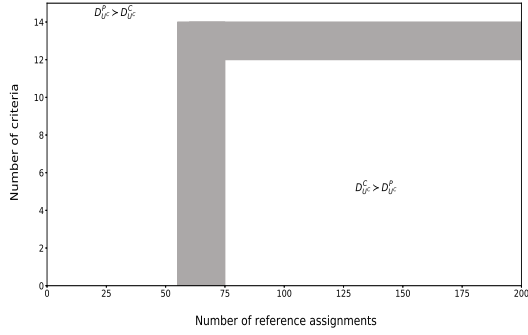


Figure 29: Computation time of SAT problems by number of reference assignments and number of criteria (3 categories) to learn a U^C model

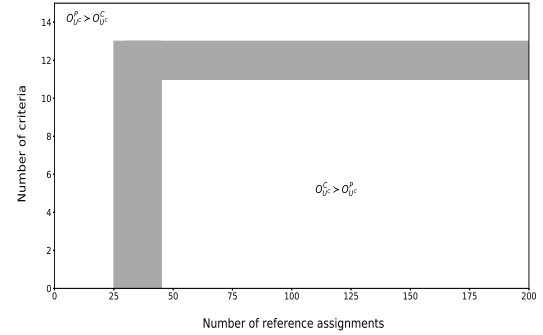


Figure 30: Computation time of MaxSAT problems by number of reference assignments and number of criteria (3 categories and 10% noise) to learn a U^C model

These two effects leads to distinguish configurations (depending on the number of criteria, and size of the learning set) in which either of two representation (coalition-based or separation-based) is faster. For a decision problem with 3 categories (and 10% noise for MaxSAT instances), Figures 27 and 28 (Figures 29 and 30, respectively) depicts which of the two representation (coalition-based or separation-based) is faster to learn U^B model (U^C model, respectively) depending on the number of alternatives and number of criteria (both for SAT and MaxSAT). These Figures offer insights to choose the appropriate representation according to the number of criteria and the number of alternatives.

Ability to restore the learning set:

As expected, all SAT instances (without noise) are able to fully restore the learning sets; this result is an experimental validation of the theoretical work developed in Section 4. Moreover, when learning a model from noisy learning sets (MaxSAT extension), we were able to infer NCS models with a restoration rate over $1 - x$, where x denotes the noise level in the learning set.

Ability to generalize:

In terms of generalization (the alignment between the output model with the ground truth), for both U^B and U^C models, coalition-based and separation-based strategies behave in analogously:

- an increase of the size of the learning set induces an improvement of the generalization index; such improvement occurs whatever the noise level (up to 20%). This means that it seems always possible to “capture the ground truth” with a sufficiently large learning set,
- an increase in the reference set noise level require a larger learning set to keep the same generalization level. This implies that the “quality” of the learning set, have a significant impact on the required size of this learning set.

6.4.2 Insights for the decision analyst

An interesting aspect of the empirical results lies in the possibility to derive insights on how to put the proposed learning algorithms in practice in an decision-aiding case study.

Defining the size of the learning set for a given number of criteria:

An important question for a decision analyst concerns the number of assignment examples to collect in order to accurately capture the DM’s preferences. Our experiments provide figures to answer such questions. In a decision problem involving 3 categories and 5 criteria, if the analyst wishes to obtain an U^B model with target level of 90% for the generalization index, and postulates a error rate of 10% in the set of assignment examples, Figure 23 informs us that the size of the learning set should be in the interval [64, 128].

Choosing the fastest formulation depending on the number of criteria and size of the learning set:

Another relevant question concerns which of the coalition-based or separation-based representation provides the lowest computing time for a given size of learning set (and number of criteria).

For a given number of criteria and for learning a U^B model, Figures 27 and 28 depict the approximate thresholds in terms of number of reference assignments from which the coalition-based representation becomes faster than the separation-based one. In the case where the preference information is perfect and for less than ~ 50 examples, the separation-based representation is faster than the coalition-based representation, and the generalization is equivalent for both representations. For MaxSAT instances, for more than ~ 50 examples and less than ~ 11 criteria, the coalition-based representation formulation is faster than the separation-based one. However, for a number of criteria exceeding ~ 13 or for less than ~ 50 reference assignments, the separation-based representation is faster. For all configurations, the separation-based representation generalizes better.

For a given number of criteria and for learning a U^C model, Figures 29 and 30 depict the approximate thresholds in terms of number of reference assignments from which the coalition-based representation becomes faster than the separation-based one. In the case where the preference information is noiseless and for more than ~ 14 criteria or for less than ~ 64 reference assignments, the separation-based representation is more efficient than the coalition-based one in terms of the computation time and the generalization index.

7 Conclusion

In this paper, we consider the multiple criteria Non-Compensatory Sorting model and its variants with a unique profile (U^B) and a unique set of sufficient coalitions (U^C). Learning this model has already been addressed by the literature, and solved by the resolution of a MIP [35] or via a specific heuristic [50] [51]. Recently, two SAT representations (coalition-based, and separation-based) have been proposed to learn such a model from perfect preference information and already proved to be superior to other approaches, see [6]. The separation-based representation was originally described in [6] but only focusing on the case with two categories. We consider in this work the generalization of this formulation to the multiple categories case for learning NCS and its variant U^B -NCS and U^C -NCS. The separation-based representation is more compact than the coalition-based one as it handles explicitly a set of sufficient coalitions that lies in the power set of the criteria. In order to handle the inconsistency in preference information, we extend the two SAT problems using MaxSAT language. Thus, for each variant of NCS, we proposed two MaxSAT programs to compute the model's parameters from noisy preference information.

The separation-based representation proposed for learning U^B and U^C models is at least as good as the coalition-based one in terms of generalization and for both types of preference information (perfect and not-so-perfect preferences). Computation time of the two representations evolves depending on the number of reference alternatives and the number of criteria; the separation-based representation performs better when the number of criteria increases, while it is not the case when the number of reference alternatives increases. Increasing the number of categories penalizes the separation-based representation proposed for learning U^B model, since the number of clauses depends quadratically on the number of categories.

However, for real world decision problems, assuming that the number of reference assignments is ~ 100 examples, we can consider two types of applications: an application that involves a large number of criteria ($|\mathcal{N}| > \sim 12$) and therefore the separation-based representation seems better as it is faster and generalizes better than the first one, and an application that involves a limited number of criteria ($|\mathcal{N}| < \sim 10$), in this case, the coalition-based representation is slightly faster and generalizes less than the separation-based one.

Finally, our work shows that, when learning MCDA models from preference information, SAT and MaxSAT languages can be relevant and efficient. This is specifically the case for ordinal MCDA aggregation procedures based on pairwise comparison of alternatives (so called outranking methods, see [22]). We believe that our work opens avenue for further research to develop new algorithms to learn outranking models from preference statements using SAT/MaxSAT language.

References

- [1] J. Almeida-Dias, J.R. Figueira, and B. Roy, ‘Electre Tri-C: A multiple criteria sorting method based on characteristic reference actions’, *European Journal of Operational Research*, **204**(3), 565–580, (2010).
- [2] J. Almeida-Dias, J.R. Figueira, and B. Roy, ‘A multiple criteria sorting method where each category is characterized by several reference actions: The Electre Tri-nC method’, *European Journal of Operational Research*, **217**(3), 567–579, (2012).
- [3] S. G. Arcidiacono, S. Corrente, and S. Greco, ‘Robust stochastic sorting with interacting criteria hierarchically structured’, *European Journal of Operational Research*, **292**(2), 735–754, (2021).
- [4] K. Belahcene, C. Labreuche, N. Maudet, V. Mousseau, and W. Ouerdane, ‘A model for accountable ordinal sorting’, in *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*, pp. 814–820, (2017).
- [5] K. Belahcene, C. Labreuche, N. Maudet, V. Mousseau, and W. Ouerdane, ‘An efficient SAT formulation for learning multiple criteria non-compensatory sorting rules from examples’, *Computers & Operations Research*, **97**, 58–71, (2018).
- [6] K. Belahcene, C. Labreuche, N. Maudet, V. Mousseau, W. Ouerdane, and Y. Chevaleyre, ‘Accountable approval sorting’, in *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI 2018)*, (2018).
- [7] K. Belahcene, V. Mousseau, M. Pirlot, and O. Sobrie, ‘Preference elicitation and learning in a multiple criteria decision aid perspective’, Technical report, Laboratoire Génie Industriel, CentraleSupélec, (February 2017). Research report 2017-02.
- [8] N. Benabbou, P. Perny, and P. Viappiani, ‘A regret-based preference elicitation approach for sorting with multicriteria reference profiles’, in *Proceeding of the 2nd workshop from multiple criteria Decision aid to Preference Learning (DA2PL)*, (2016).
- [9] N. Benabbou, P. Perny, and P. Viappiani, ‘Incremental elicitation of Choquet capacities for multicriteria choice, ranking and sorting problems’, *Artificial Intelligence*, **246**, 152–180, (2017).
- [10] J. Błaszczynski, S. Greco, and R. Słowiński, ‘Multi-criteria classification – a new scheme for application of dominance-based decision rules’, *European Journal of Operational Research*, **181**(3), 1030 – 1044, (2007).
- [11] D. Bouyssou and T. Marchant, ‘An axiomatic approach to noncompensatory sorting methods in MCDM, I: the case of two categories’, *European Journal of Operational Research*, **178**(1), 217–245, (2007).
- [12] D. Bouyssou and T. Marchant, ‘An axiomatic approach to noncompensatory sorting methods in MCDM, II: more than two categories’, *European Journal of Operational Research*, **178**(1), 246–276, (2007).
- [13] D. Bouyssou, T. Marchant, and M. Pirlot, ‘A theoretical look at ELECTRE TRI-nB’. working paper, 2020.
- [14] A. Chateauneuf and J.-Y. Jaffray, ‘Derivation of some results on monotone capacities by mobius inversion’, in *Uncertainty in Knowledge-Based Systems*, eds., B. Bouchon and R. R. Yager, pp. 95–102. Springer, (1987).
- [15] S. Corrente, M. Doumpos, S. Greco, R. Słowiński, and C. Zopounidis, ‘Multiple criteria hierarchy process for sorting problems based on ordinal regression with additive value functions’, *Annals of Operations Research*, **251**(1), 117–139, (2017).
- [16] J.M. Devaud, G. Groussaud, and E. Jacquet-Lagrez, ‘UTADIS: Une methode de construction de fonctions d’utilite additives rendant compte de jugements globaux’, in *European working group on MCDA, Bochum , Germany*, (1980).
- [17] M. Doumpos and C. Zopounidis, *Multicriteria Decision Aid Classification Methods*, Springer, 2002.

- [18] A. Fallah Tehrani, W. Cheng, and E. Hüllermeier, ‘Choquistic regression: Generalizing logistic regression using the Choquet integral’, in *in Proceedings EUSFLAT-2011 7th International Conference on the European Society for Fuzzy Logic and Technology*, p. 868–875, (2011).
- [19] E. Fernández, J.R. Figueira, and J. Navarro, ‘An indirect elicitation method for the parameters of the Electre Tri-nB model using genetic algorithms’, *Applied Soft Computing*, **77**, 723–733, (2019).
- [20] E. Fernández, J.R. Figueira, J. Navarro, and B. Roy, ‘Electre tri-nB: A new multiple criteria ordinal classification method’, *European Journal of Operational Research*, **263**(1), 214–224, (2017).
- [21] J. Figueira, S. Greco, B. Roy, and R. Słowiński, ‘ELECTRE methods: Main features and recent developments’, in *Handbook of Multicriteria Analysis*, eds., C. Zopounidis and P.M. Pardalos, 51–89, Springer, (2010).
- [22] J. Figueira, V. Mousseau, and B. Roy, ‘Electre methods’, in *Multiple criteria decision analysis: State of the art surveys*, 133–153, Springer, (2005).
- [23] J. Furnkranz and E. Hullermeier, *Preference Learning*, Springer, 2011.
- [24] S. Greco, B. Matarazzo, and R. Slowinski, ‘Rough sets theory for multicriteria decision analysis’, *European Journal of Operational Research*, **129**(1), 1 – 47, (2001).
- [25] S. Greco, V. Mousseau, and R. Slowinski, ‘Multiple criteria sorting with a set of additive value functions’, *European Journal of Operational Research*, **207**(3), 1455–1470, (Dec. 2010).
- [26] E. Herrera-Viedma, F. Herrera, F. Chiclana, and M. Luque, ‘Some issues on consistency of fuzzy preference relations’, *European Journal of Operational Research*, **154**(1), 98 – 109, (2004).
- [27] E. Jacquet-Lagrange and J. Siskos, ‘Assessing a set of additive utility functions for multicriteria decision-making, the uta method’, *European Journal of Operational Research*, **10**(2), 151–164, (1982).
- [28] M. Kadzinski and M. Martyn, ‘Enriched preference modeling and robustness analysis for the Electre Tri-B method’, *Annals of Operations Research*, 1–35, (2020).
- [29] M. Kadziński and K. Ciomek, ‘Active learning strategies for interactive elicitation of assignment examples for threshold-based multiple criteria sorting’, *European Journal of Operational Research*, (2021).
- [30] M. Kadziński, M. Ghaderi, and M. Dabrowski, ‘Contingent preference disaggregation model for multiple criteria sorting problem’, *European Journal of Operational Research*, **281**(2), 369–387, (2020).
- [31] M. Kadziński, S. Greco, and R. Słowiński, ‘Robust ordinal regression for dominance-based rough set approach to multiple criteria sorting’, *Information Sciences*, **283**, 211–228, (2014).
- [32] M. Kadziński, T. Tervonen, and J.R. Figueira, ‘Robust multi-criteria sorting with the outranking preference model and characteristic profiles’, *Omega*, **55**, 126 – 140, (2015).
- [33] M. Köksalan and S. Ozpeynirci, ‘An interactive sorting method for additive utility functions’, *Computers & Operations Research*, **36**(9), 2565 – 2572, (2009).
- [34] C. Labreuche, ‘A general framework for explaining the results of a multi-attribute preference model’, *Artificial Intelligence*, **175**(7), 1410–1448, (2011).
- [35] A. Leroy, V. Mousseau, and M. Pirlot, ‘Learning the parameters of a multiple criteria sorting method’, in *International Conference on Algorithmic Decision Theory*, pp. 219–233. Springer, (2011).
- [36] J. Liu, M. Kadzinski, X. Liao, and X. Mao, ‘Data-driven preference learning methods for value-driven multiple criteria sorting with interacting criteria’, *INFORMS Journal of Computing*, **to appear**, (2020).

- [37] J. Liu, X. Liao, M. Kadziński, and R. Słowiński, ‘Preference disaggregation within the regularization framework for sorting problems with multiple potentially non-monotonic criteria’, *European Journal of Operational Research*, **276**(3), 1071–1089, (2019).
- [38] J. Liu, X. Liao, X. Mao, Y. Wang, and M. Kadzinski, ‘A preference learning framework for multiple criteria sorting with diverse additive value models and valued assignment examples’, *European Journal of Operational Research*, **286**(3), 963–985, (2020).
- [39] J.-L. Marichal, P. Meyer, and M. Roubens, ‘Sorting multi-attribute alternatives: The TOMASO method’, *Computers & Operations Research*, **32**(4), 861 – 877, (2005).
- [40] P. Minoungou, V. Mousseau, W. Ouerdane, and Scotton P., ‘Learning an MR-Sort model from data with latent criteria preference directions’, in *Proceeding of the 2nd workshop from multiple criteria Decision aid to Preference Learning (DA2PL)*, (2020).
- [41] V. Mousseau, L.C. Dias, J. Figueira, C. Gomes, and J.N. Clímaco, ‘Resolving inconsistencies among constraints on the parameters of an MCDA model’, *European Journal of Operational Research*, **147**(1), 72–93, (2003).
- [42] V. Mousseau and R. Słowiński, ‘Inferring an ELECTRE TRI model from assignment examples’, *Journal of global optimization*, **12**(2), 157–174, (1998).
- [43] P. Perny, ‘Multicriteria filtering methods based on concordance and non-discordance principles’, *Annals of Operations Research*, **80**, 137–165, (1998).
- [44] B. Roy, ‘The outranking approach and the foundations of Electre methods’, *Theory and Decision*, **31**(1), 49–73, (1991).
- [45] B. Roy, *Multicriteria Methodology for Decision Aiding*, Kluwer Academic, Dordrecht, 1996.
- [46] C. Rudin and S. Ertekin, ‘Learning customized and optimized lists of rules with mathematical programming’, *Mathematical Programming Computation*, **10**, 659–702, (2018).
- [47] Y. Siskos, E. Grigoroudis, and N. Matsatsinis, ‘UTA Methods’, in *Multiple Criteria Decision Analysis*, eds., S. Greco, M. Ehrgott, and J. Figueira, International Series in OR/MS, 315–362, Springer, (2016).
- [48] O. Sobrie, *Learning preferences with multiple-criteria models*, Ph.D. dissertation, Université de Mons (Faculté Polytechnique) and Université Paris-Saclay (CentraleSupélec).
- [49] O. Sobrie, M.A. Lazouni, S. Mahmoudi, V. Mousseau, and M. Pirlot, ‘A new decision support model for preanesthetic evaluation’, *Computer Methods and Programs in Biomedicine*, **133**, 183–193, (2016).
- [50] O. Sobrie, V. Mousseau, and M. Pirlot, ‘Learning a majority rule model from large sets of assignment examples’, in *Algorithmic Decision Theory*, eds., P. Perny, M. Pirlot, and A. Tsoukiás, volume 8176 of *Lecture Notes in Artificial Intelligence*, pp. 336–350. Springer, (2013).
- [51] O. Sobrie, V. Mousseau, and M. Pirlot, ‘Learning the parameters of a non compensatory sorting model’, in *Algorithmic Decision Theory*, ed., T. Walsh, volume 9346 of *Lecture Notes in Artificial Intelligence*, pp. 153–170, Lexington, KY, USA, (2015). Springer.
- [52] O. Sobrie, V. Mousseau, and M. Pirlot, ‘Learning monotone preferences using a majority rule sorting model’, *International Transactions in Operational Research*, **26**(5), 1786–1809, (2019).
- [53] N. Sokolovska, Y. Chevaleyre, and J.-D. Zucker, ‘A provable algorithm for learning interpretable scoring systems’, in *International Conference on Artificial Intelligence and Statistics, AISTATS 2018*, eds., A. J. Storkey and F. Pérez-Cruz, pp. 566–574, (2018).

- [54] T. Tervonen, J. Figueira, R. Lahdelma, and J. and P. Salminen Almeida Dias, ‘A stochastic method for robustness analysis in sorting problems’, *European Journal of Operational Research*, **192**(1), 236–242, (2009).
- [55] B. Ustun and C. Rudin, ‘Supersparse linear integer models for optimized medical scoring systems’, *Machine Learning*, **102**(3), 349–391, (2016).
- [56] J. Zheng, S. Metchebon Takougang, V. Mousseau, and M. Pirlot, ‘Learning criteria weights of an optimistic Electre Tri sorting rule’, *Computers & Operations Research*, **49**, 28 – 40, (2014).

Appendices

A SAT formulations for NCS variants with more than 2 categories

A.1 Learning a U^B -NCS Model

When trying to fit a U^B -NCS model, neither a variables nor s variables are indexed by exigence level; s variables are indexed by a criterion i and a pair of alternatives $g, b \in \mathbb{X}^*$ such that g is preferred to b , i.e. $\alpha(g) \succ \alpha(b)$.

The propositional formula obtained by following the representation strategy based on the pairwise separation of alternatives is particularly simple and elegant.

Definition A.1. Given an instance of *Inv-NCS* with an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$, we define the boolean function $\Phi_{\alpha}^{P'B}$ with variables $\langle a_{i,x} \rangle_{i \in \mathcal{N}, x \in \mathbb{X}^*}$ and $\langle s_{i,g,b} \rangle_{i \in \mathcal{N}, \alpha(g) \succ \alpha(b)}$, as the conjunction of clauses:

$$\begin{aligned} \Phi_{\alpha, U^B\text{-NCS}}^{P'B} &= \phi_{\alpha}^{P'B1} \wedge \phi_{\alpha}^{P'B3} \wedge \phi_{\alpha}^{P'B4} \wedge \phi_{\alpha}^{P'B5} \\ \phi_{\alpha}^{P'B1} &= \bigwedge_{i \in \mathcal{N}} \bigwedge_{x' \succ_i x \in \mathbb{X}^*} (a_{i,x'} \vee \neg a_{i,x}) \\ \phi_{\alpha}^{P'B3} &= \bigwedge_{i \in \mathcal{N}} \bigwedge_{\alpha(g) \succ \alpha(b)} (\neg s_{i,g,b} \vee \neg a_{i,b}) \\ \phi_{\alpha}^{P'B4} &= \bigwedge_{i \in \mathcal{N}} \bigwedge_{\alpha(g) \succ \alpha(b)} (\neg s_{i,g,b} \vee a_{i,g}) \\ \phi_{\alpha}^{P'B5} &= \bigwedge_{\alpha(g) \succ \alpha(b)} (\bigvee_{i \in \mathcal{N}} s_{i,g,b}) \end{aligned}$$

Corollary A.1. Given a context, an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$ can be represented in the *Non-Compensatory sorting model* with unique profile if, and only if $\Phi_{\alpha}^{P'B}$ is satisfiable.

This condition is obviously necessary. It is sufficient because the sets of observably sufficient and insufficient coalitions are nested by construction, even in the case $\mathcal{A}_i^2 = \dots = \mathcal{A}_i^p$.

A.2 Learning a U^C -NCS Model

We describe here the generalization of the pairwise separation formulation Φ_{α}^P (see Definition 4.3) to the multiple category case for fitting a U^C -NCS (Unique set of sufficient coalitions) model. Given a nesting of approved sets $\langle \mathcal{A}_i^k \rangle$, this unique set of sufficient coalitions satisfies all the constraints put by the observed sufficient and insufficient coalitions of criteria at every exigence level. This observation yields the following lower and upper bounds:

$$\begin{aligned} \mathcal{S}_{\langle \mathcal{A}_i^k \rangle}(\alpha) &= Cl_{\mathcal{P}(\mathcal{N})}^{\supseteq} \left(\bigcup_{k \in [2..p]} \bigcup_{g \in \alpha^{-1}(C^{\succeq k})} \{i \in \mathcal{N} : g \in \mathcal{A}_i^k\} \right) \\ \mathcal{F}_{\langle \mathcal{A}_i^k \rangle}(\alpha) &= Cl_{\mathcal{P}(\mathcal{N})}^{\subseteq} \left(\bigcup_{k \in [2..p]} \bigcup_{b \in \alpha^{-1}(C^{\prec k})} \{i \in \mathcal{N} : b \in \mathcal{A}_i^k\} \right) \end{aligned}$$

In turn, this entails a modification of the third condition (pairwise separation) of the representation theorem (Theorem 4.2):

- 3C. (*pairwise separation for a unique set of sufficient coalitions*) for each exigence levels $k \in [2..p]$ and $k' \in [2..p]$, for each pair of alternatives $(g, b) \in (\mathbb{X}^*)^2$ such that $g \in \alpha^{-1}(C^{\succeq k'})$ and $b \in \alpha^{-1}(C^{\prec k})$, there is at least one point of view $i \in \mathcal{N}$ such that $g \in \mathcal{A}_i^{k'}$ and $b \notin \mathcal{A}_i^k$.

We translate this modified representation theorem into a SAT formulation equisatisfiable with Inv- U^B -NCS, using variables a indexed by a criterion, an exigence level and a reference alternative, as well as variables s indexed by a criterion, a pair of exigence levels, and a pair of alternatives.

Definition A.2. Given an instance of Inv-NCS with an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$, we define the boolean function $\Phi_\alpha^{P'C}$ with variables $\langle a_{i,k,x} \rangle_{i \in \mathcal{N}, k \in \{2..p\}, x \in \mathbb{X}^*}$ and $\langle s_{i,k,k',g,b} \rangle_{i \in \mathcal{N}, k \in \{2..p\}, k' \in \{2..p\}, g \in C^{\succeq k'}, b \notin C^{\succeq k}}$, as the conjunction of clauses:

$$\Phi_\alpha^{P'C} = \phi_\alpha^{P'1} \wedge \phi_\alpha^{P'2} \wedge \phi_\alpha^{P'C3} \wedge \phi_\alpha^{P'C4} \wedge \phi_\alpha^{P'C5}$$

$$\phi_\alpha^{P'C3} = \bigwedge_{i \in \mathcal{N}, k \in [2..p], k' \in [2..p]} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\neg s_{i,k,k',g,b} \vee \neg a_{i,k,b})$$

$$\phi_\alpha^{P'C4} = \bigwedge_{i \in \mathcal{N}, k \in [2..p], k' \in [2..p]} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\neg s_{i,k,k',g,b} \vee a_{i,k',g})$$

$$\phi_\alpha^{P'C5} = \bigwedge_{k \in [2..p], k' \in [2..p]} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\bigvee_{i \in \mathcal{N}} s_{i,k,k',g,b})$$

Formulations of $\phi_\alpha^{P'1}$ and $\phi_\alpha^{P'2}$ can be found in Def. 4.4.

Corollary A.2. Given a context, an assignment $\alpha : \mathbb{X}^* \rightarrow \{C^1 \prec \dots \prec C^p\}$ can be represented in the Non-Compensatory sorting model with a unique set of sufficient coalitions if, and only if $\Phi_\alpha^{P'C}$ is satisfiable.

B MaxSAT relaxations based on pairwise separation conditions for more than two categories

We provide here extensions of the MaxSAT formulation presented in Section 5.2, to the case with multiple categories. They rely on the fact that an NCS model with p categories is informally the combination of $p - 1$ NCS models with two categories whose parameters satisfy the nesting conditions on coalitions and satisfactory values. The maximization of the restoration in the second formulation is equivalent to the simultaneous maximization of the restoration in the sub-problems with two categories. On top of the ‘ z ’ variables encoding the correct restoration of a reference alternative, we introduce intermediate switches:

- ‘ y ’ variables, indexed by an alternative $x \in \mathbb{X}^*$ and an exigence level $k \in [2..p]$, encode the proper restoration of alternative x by the 2-categories NCS model with $\text{GOOD} = C^{\succeq k}$ and $\text{BAD} = C^{\prec k}$.

These variables are logically tied to the ‘ z ’ variables by the following conjunction of hard clauses:

$$\phi_\alpha^{\widetilde{P'}yz} = \bigwedge_{x \in \mathbb{X}^*} \bigwedge_{k \in [2..p]} (y_{k,x} \vee \neg z_x)$$

While the objective in the MaxSAT formulation is to maximize the number of properly classified alternatives, this goal is reached by the simultaneous maximization of the restoration rate in each sub-problem with two categories, leading to the introduction of a number of sub-goals:

$$\phi_\alpha^{\text{subgoals}} = \bigwedge_{k \in [2..p]} \bigwedge_{x \in \mathbb{X}^*} y_{k,x}$$

The soft clause $\phi_\alpha^{\text{goal}}$ is given weight w_1 , and each one of the clause appearing in the conjunction $\phi_\alpha^{\text{subgoals}}$ is given weight w_2 , while the hard clauses are given weight w_{max} . These weights are chosen so that $w_{\text{max}} \gg w_1 \gg w_2$, and more precisely : $(p - 1) |\mathbb{X}^*| w_2 < w_1$; and $|\mathbb{X}^*| w_1 < w_{\text{max}}$.

The hard clauses differ according to the target model.

B.1 Learning an NCS model

Use the following conjunction of hard clauses: $\phi_\alpha^{P'1} \wedge \phi_\alpha^{P'2} \wedge \widetilde{\phi}_\alpha^{P'3} \wedge \widetilde{\phi}_\alpha^{P'4} \wedge \widetilde{\phi}_\alpha^{P'5} \wedge \widetilde{\phi}_\alpha^{P'yz}$.

$$\begin{aligned}\widetilde{\phi}_\alpha^{P'3} &= \bigwedge_{i \in \mathcal{N}, 2 \leq k \leq k' \leq p} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\neg s_{i,k,k',g,b} \vee \neg a_{i,k,b}) \\ \widetilde{\phi}_\alpha^{P'4} &= \bigwedge_{i \in \mathcal{N}, 2 \leq k \leq k' \leq p} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\neg s_{i,k,k',g,b} \vee a_{i,k',g}) \\ \widetilde{\phi}_\alpha^{P'5} &= \bigwedge_{k \in [2..p], 2 \leq k \leq k' \leq p} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\bigvee_{i \in \mathcal{N}} s_{i,k,k',g,b} \vee \neg y_{k,b} \vee \neg y_{k',g})\end{aligned}$$

Note that the conjunction $\widetilde{\phi}_\alpha^{P'3}$ (resp. $\widetilde{\phi}_\alpha^{P'4}$) subsumes the conjunction $\phi_\alpha^{P'3}$ (resp. $\phi_\alpha^{P'4}$) introduced in Def. 4.4, but that, together with the constraints $\phi_\alpha^{P'2}$, is equivalent to it. While this redundancy is not needed in the SAT formulation, it helps formulate the subgoals of the MaxSAT formulation.

B.2 Learning a U^C -NCS model

Use the following conjunction of hard clauses: $\phi_\alpha^{P'1} \wedge \phi_\alpha^{P'2} \wedge \phi_\alpha^{P'C3} \wedge \widetilde{\phi}_\alpha^{P'C4} \wedge \widetilde{\phi}_\alpha^{P'C5} \wedge \widetilde{\phi}_\alpha^{P'yz}$.

Formulas $\phi_\alpha^{P'1}$ and $\phi_\alpha^{P'2}$ are introduced in Def. 4.4, $\phi_\alpha^{P'C3}$ and $\phi_\alpha^{P'C4}$ are introduced in Def A.2, and

$$\widetilde{\phi}_\alpha^{P'C5} = \bigwedge_{k \in [2..p], k' \in [2..p]} \bigwedge_{g \in \alpha^{-1}(C^{\succeq k'}), b \in \alpha^{-1}(C^{\prec k})} (\bigvee_{i \in \mathcal{N}} s_{i,k,k',g,b} \vee \neg y_{k,b} \vee \neg y_{k',g}).$$

B.3 Learning a U^B -NCS model

As it is the case when addressing the decision problem, the U^B -NCS model can be learned with a MaxSAT formulation which is very close to the one used in the case of two categories, without using any ‘ y ’ variables. Use the following conjunction of hard clauses (each one with weight w_{max}): $\phi_\alpha^{P'B1} \wedge \phi_\alpha^{P'B3} \wedge \phi_\alpha^{P'B4} \wedge \widetilde{\phi}_\alpha^{P'B5}$, together with the soft clause ϕ_α^{goal} with weight $w_1 < w_{max}/|\mathbb{X}^*|$. Formulas $\phi_\alpha^{P'B1}$, $\phi_\alpha^{P'B3}$ and $\phi_\alpha^{P'B4}$ can be found in Def. A.1, and

$$\widetilde{\phi}_\alpha^{P'B5} = \bigwedge_{\alpha(g) \succ \alpha(b)} (\bigvee_{i \in \mathcal{N}} s_{i,g,b} \neg z_b \vee \neg z_g).$$