Representing Interval Orders by Weighted Bases: some Complexity Results

Pierre Marquis [a] Meltem Öztürk [a]

[a] *CRIL-CNRS, Université d'Artois, 62307 Lens, France*

Abstract

This paper is centered on the notion of interval order as a model for preferences. We introduce a family of representation languages for such orders, parameterized by a scale and an aggregation function. We show how interval orders can be represented by elements of those languages, called weighted bases. We identify the complexity of the main decision problems to be considered for exploiting such representations of interval orders (including the comparison problems and the non-dominance problem). We also show that our representation of interval orders based on weighted bases encompasses the penalty-based representation of complete preorders as a specific case.

Key words: Compact representation of preferences, Preferences over combinatorial domains, Computational complexity

1 Introduction

Dealing with preferences over alternatives is an important issue in many fields, like economics, decision theory, and artificial intelligence. Preferences are generally formulated as binary relations which are related to a notion of "order". Different types of models may be used for their representation. Many existing models are quantitative ones, the quantification of preferences rendering easier the search for optimal or near-optimal decisions. Much work has been devoted so far to such models in social choice theory. In the majority of these works, preference is given by a utility function (i.e., a mapping from the set of alternatives under consideration to the set $\mathbb{R}$ of real numbers). On the other hand, pure qualitative settings are adequate when quantifying preferences is meaningless [1] or when no quantification of prefer-

[1] In real life situations, when the preferences are just ordinal, the following type of affirmations about preferences intensities can be a non-sense: "the preference of alternative a over alternative b is two times the preference of alternative c over alternative d".

ences is known. In such settings, a preference relation is typically defined as a complete preorder (i.e., a reflexive and transitive relation) over the set of alternatives. However, such a model for preferences does not prove adequate to all situations, and other models (generalizing the complete preorder one) have been pointed out. In particular, a well-known problem with the complete preorder structure for preference is that the associated indifference relation is necessarily transitive, and it is known that such a property may be violated in the presence of thresholds as shown in the famous example of Luce [18]:

Example 1 *Let the set of alternatives $A = \{c(0), \ldots, c(100)\}$ consists of 101 cups of coffee which are identical except that the i^{th} cup $c(i)$ contains i grains of sugar. Any human agent who wants to compare those alternatives by tasting them is not able no make any distinction between $c(i)$ and $c(i+1)$, hence $c(i)$ and $c(i+1)$ ($i \in 0 \ldots 99$) are indifferent for the agent. Assuming such indifference relation transitive would imply that $c(0)$ and $c(100)$ are indifferent as well, which is not adequate whenever the agent prefers coffee with sugar.*

Among other relational structures like quasi-orders, tolerance orders, split semiorders, etc. (for more details, see e.g. [11]), interval orders have been introduced for handling such scenarios. Indeed, in contrast to the associated strict preference relation, the indifference relation induced by an interval order is not necessarily transitive.

Once an adequate model for preference has been chosen, the representation issue has still to be addressed. In this direction many contributions in decision theory are based on the representational theory of measurement, formalized in [23] and presented in details in the three-volume set by Krantz et al. [14], Suppes et al. [15] and Luce et al. [19]. Concerning the interval orders and semiorders (special case of the first ones), the axiomatic analysis of what is called “interval orders” has been given by Wiener [27], then the term “semiorders” has been introduced by Luce [18] and many results about their representations have been obtained by different researchers (for more details see [10], [22]). In the classical numerical representation of interval orders, an interval (with a uniform length in the case of semiorders) is associated to each alternative and each alternative is said to be preferred to another one if and only if its associated interval is completely to the right of the other’s interval. In the following, this is what we call an interval representation of an interval order. Doignon [7] has observed that it is possible to define a minimal interval representation of interval orders and Fishburn [10] has been interested in another optimization problem about interval orders (minimal number of different interval lengths). Isaak [13] has obtained a minimax theorem for the minimum number of end points in an interval representation of an interval order by the help of potentials in digraphs.

Unfortunately, when the set of alternatives is huge, neither the explicit representation of the interval order (i.e., its representation by pairs) nor its interval representations are feasible; especially when the set of alternatives has a combinatorial structure, i.e., it is a cartesian product of a finite domain for each one of a set of variables. For instance, let us imagine that a decision maker is trying to choose a new computer. A set of variables may be hard disk, ram, screen size, sound card, DVD reader and DVD writer. In this case the set of alternatives is all the possible combinations of the evaluations of these variables. In such a situation the space of possible alternatives has a size exponential in the number of variables and it is therefore not feasible (for space reasons) to associate one interval to each alternative. Hence, it is important to define more compact representation languages for interval orders and to evaluate them.

Generally speaking, by a *preference relation* over a set A of alternatives, we mean a binary relation over A. By a *representation language* for preference relations, we mean a set of symbolic descriptions of such relations, and some procedures for exploiting them. For the sake of generality, we do not put any strong restrictions on the acceptable representations, except that they must be finite. Thus, within a representation language, each element of the preference structure is represented by a *word*. We define a representation language through an *alphabet* which can be any finite set of symbols; e.g. a language of first-order logic is often expressed using an alphabet which, besides logical symbols such as connectives $\wedge$, $\vee$, $\neg$ and quantifiers $\forall$, $\exists$ contains elements $x_0, x_1, x_2, \ldots$ playing the role of variables. A word over an alphabet can be any finite sequence of elements of the alphabet. And finally a language over an alphabet is just a subset of the set of all words over the alphabet.

For the purpose of evaluating a representation language for preference relations, the following three criteria are of great value:

- Simplicity and modularity: a representation language is expected to have a simple (yet formal) semantics expressing the connection between the representation and the corresponding (explicit) preference relation; such a connection must be easy to understand. Modularity is the ability to specify the preference relation within the representation language in a piecewise way.
- Complexity issues: they indicate the computational effort which must be spent to realize a number of treatments of interest on the preference relation represented in the chosen language. Such treatments (vote, aggregation, etc.) typically depend on the way preferences have to be exploited in the application under consideration, and are often based on some basic queries and transformations. Some basic queries consist in determining whether a given alternative is preferred or not to another and determining whether a given alternative is undominated. The focus is typically laid on worst case scenarios.
- Expressiveness and spatial efficiency: expressiveness gives the (relative) aptitude of a representation language to encode a family of preference relations (total order, preorder, partial order, etc.), while spatial efficiency gives its aptitude to do it using little space (it refines expressiveness); both notions are formalized as preorders on the set of all representation languages (see e.g. [5]):

Definition 1 *A representation language L_1 for preference relations is said to be* at least as expressive as *a representation language L_2 if and only if every preference relation which can be represented in L_2 can also be represented in L_1.*

Definition 2 *A representation language L_1 for preference relations is said to be* at least as succinct as *a representation language L_2 if and only if there exists a polynomial p such that every preference relation which can be represented by an element r_2 in L_2 can be represented by an element r_1 in L_1 such that $|r_1| \leq p(|r_2|)$.*[2]

Since the complexity of any algorithm is a function relating its input size with the amount of resources (time or space) needed to achieve the computation, the complexity results for treatments based on a given representation language must be interpreted in light of its spatial efficiency.

The first criterion (simplicity and modularity) is fundamental in the direction of preference elicitation and other human-computer interactions about the preference relation. It shows the perspective of preference representation more general than preference compilation, which mainly consists in turning a given preference representation into another one, so as to optimize at least one of the last two criteria. For instance, a preference relation (viewed as a binary relation over a set of n alternatives) can be compactly represented as a Boolean function with $2\lceil log_2 n\rceil$ arguments, where the first $\lceil log_2 n\rceil$ bits encodes a first alternative a_1 and the last $\lceil log_2 n\rceil$ bits encodes a second alternative a_2 and the function takes the value 1 if and only if a_1 is at least as preferred as a_2; the Boolean function itself can be represented in many different ways (CNF formulae, Binary Decision Diagrams, etc., see e.g. [6]). The encoding procedure and/or the definition of the Boolean function (since it may be complicated) may prevent the resulting data structure from achieving what is expected from the point of view of simplicity and modularity. For instance, the following formula $((\neg x_1 \vee x_2) \wedge (x_1 \vee \neg x_2)) \vee ((\neg x_1 \vee \neg x_2) \wedge (\neg x_1 \vee (x_3 \wedge \neg x_4)) \wedge ((x_2 \wedge (x_3 \Leftrightarrow \neg x_4)) \vee (x_1 \wedge \neg x_2 \wedge x_3 \wedge \neg x_4)))$ can be viewed as a representation of the preorder $R = \{(a,a),(a,b),(b,a),(b,b),(a,c),(b,c),(a,d),(b,d),(c,d),(c,c),(d,d)\}$ over $A = \{a,b,c,d\}$ with the encoding $a = 00, b = 11, c = 01, d = 10$ [3]. Clearly enough, the connection between R and the formula representing it is not so salient.

While much effort has been devoted to the representation issue for utility functions or preorders (complete or partial) for the last years (see among others [9,1–3,12,17,4]), the compact representation of interval orders has not been addressed so far (as far as we know).

[2] The size $|r|$ of any word r is the number of symbols in it.

[3] For instance we have $(a,b) \in R$ since the assignment $x_1 = x_2 = false$ and $x_3 = x_4 = true$ satisfies the logical proposition

In this paper, we contribute to fill this gap by showing how interval orders can be compactly encoded by weighted bases, i.e., multisets of propositional formulae associated to intervals over a scale. In order to handle a number of different scenarios and achieve some flexibility, the scale and an aggregation function over this scale are considered as parameters in our framework; each choice of a scale and aggregation function gives rise to a specific representation language. We give a simple formal semantics for such languages and show them modular. As to the expressiveness issue, we explain how any interval order over a finite set of alternatives can be represented as a weighted base in some representation languages. We also show that a language of weighted bases is strictly more expressive than the language of penalty-based representations of complete preorders, as considered in [16,5,17,4]. As to the spatial efficiency issue, we show that our representation languages are strictly more compact than the language of explicit representations and the language of interval representations of such orders. Then we investigate the complexity of a number of decision problems pertaining to the exploitation of interval orders represented by weighted bases. We show that several key decision problems (comparing alternatives, determining whether an alternative is feasible) remain tractable when interval orders are represented by weighted bases, while some other key decision problems (determining whether an alternative is undominated) become "mildly" hard (i.e., at the first level of the polynomial hierarchy); this appears as the price to be paid for the gain in spatial efficiency offered by our representation languages. Interestingly, our results show that in many cases the additional expressive power offered by our approach does not lead to a complexity shift, compared to the penalty-based approach to complete preorders representation.

The rest of the paper is organized as follows. Some formal preliminaries are given in Section 2. How interval orders can be represented in a simple, modular way by weighted bases is shown in Section 3. The expressiveness and spatial efficiency issues are addressed in Section 4. Complexity results are provided in Section 5. Section 6 concludes the paper and gives some perspectives for further research.

2 Formal Preliminaries

2.1 Propositional logic

We consider a propositional language $PROP_{PS}$ generated in the usual way from a finite set PS of propositional atoms, the connectives $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$, and the Boolean constants $true$ (verum), $false$ (falsum). For every formula Σ from $PROP_{PS}$, $Var(\Sigma)$ denotes the set of all atoms from PS occurring in Σ. The size $|\Sigma|$ of any formula Σ is the number of symbols (atoms and connectives) used to write it.

A world over PS is a total function (i.e., a mapping) ω from PS to $\{0, 1\}$, which can be represented as a bit vector (once a total, strict ordering over PS has been specified). For instance, if $PS = \{x, y, z\}$ ordered in this way, then 101 represents the world ω such that $\omega(x) = 1, \omega(y) = 0$ and $\omega(z) = 1$. The set of all worlds is denoted by 2^{PS}.

The notion of satisfaction is defined in the standard truth functional way. When a world ω satisfies a formula ϕ, we write $\omega \models \phi$ and say that ω is a model of ϕ. $Mod(\phi)$ denotes the set of all models of ϕ. Inference is defined as model containment, à la Tarski. $\equiv$ denotes logical equivalence.

2.2 Binary relations, scales and intervals

Let A be a set and $R \subseteq A \times A$ be a relation over A; we consider:

- The relation I_R over A defined by $\forall a_1, a_2 \in A$, $(a_1, a_2) \in I_R$ if and only if $(a_1, a_2) \in R$ and $(a_2, a_1) \in R$ (note that I_R is the symmetric part of R).
- The relation P_R over A defined by $\forall a_1, a_2 \in A$, $(a_1, a_2) \in P_R$ if and only if $(a_1, a_2) \in R$ and $(a_2, a_1) \notin R$ (hence P_R is the asymmetric part of R).

Clearly enough, we have $R = I_R \cup P_R$ and $I_R \cap P_R = \emptyset$.

By construction, I_R is a symmetric relation and P_R an asymmetric one.[4] If R is interpreted as a preference relation such that $(a_1, a_2) \in R$ if and only if a_1 is at least as preferred as a_2, then I_R is the associated indifference relation ($(a_1, a_2) \in I_R$ if and only if a_1 and a_2 are equally preferred) and P_R is the associated strict preference relation ($(a_1, a_2) \in P_R$ if and only if a_1 is strictly preferred to a_2).

A scale S is a totally ordered set which has a least element $\bot$ and a greatest element $\top$ such that $\top \neq \bot$. $\leq$ denotes the corresponding (complete) order. Let $=$ denote the identity relation over S. We denote by $<, >, \geq$ the binary relations over S given respectively by $s_1 < s_2$ if and only if $s_1 \leq s_2$ and not ($s_1 = s_2$), $s_1 > s_2$ if and only if not ($s_1 \leq s_2$), and $s_1 \geq s_2$ if and only if not ($s_1 < s_2$), whatever the elements s_1 and s_2 of S.

The *set of all intervals* over S is Int_S given by

$$Int_S = \{[s_1, s_2], [s_1, s_2), (s_1, s_2], (s_1, s_2) \mid s_1, s_2 \in S\},$$

where:

- $[s_1, s_2]$ is a notation for $\{s \in S \mid s_1 \leq s \leq s_2\}$;
- $[s_1, s_2)$ is a notation for $\{s \in S \mid s_1 \leq s < s_2\}$;
- $(s_1, s_2]$ is a notation for $\{s \in S \mid s_1 < s \leq s_2\}$;

[4] A binary relation P is asymmetric iff $\forall a, b \in A$, if $(a, b) \in P$, then $(b, a) \notin P$

- (s_1, s_2) is a notation for $\{s \in S \mid s_1 < s < s_2\}$.

For any i of Int_S of the form $i = [s_1, s_2]$, $[s_1, s_2)$, $(s_1, s_2]$, or (s_1, s_2), we note $l(i) = s_1$ and $u(i) = s_2$. $lc(i)$ (resp. $rc(i)$) is the proposition stating that i is left-closed (resp. right-closed), i.e., of the form $[s_1, s_2]$ or $[s_1, s_2)$ (resp. of the form $[s_1, s_2]$ or $(s_1, s_2]$).

Each of the pairs $[s_1, s_2]$, $[s_1, s_2)$, $(s_1, s_2]$, (s_1, s_2) must be considered as a concise *notation* for a (possibly infinite) set. However, a given interval over a scale S (viewed as a connected subset i of S, i.e., a subset of S such that every element of S lying between two elements of the subset belons to the subset as well) may easily have more than one pair representation. This is rather obvious for the empty interval (i.e., the empty set) which can be represented by any pair where $s_1 > s_2$, but also for non-empty intervals in some cases. For instance, if S is the set of integers, its subset $\{3, 4, 5\}$ can be represented by the pair $(2, 6)$ but also by the pair $[3, 5]$.

In order to avoid any ambiguity, we can associate to every pair i representing an interval over a scale S a *normal representation* $i\downarrow$. This calls for the notions of successor and predecessor. Let x and y two elements of S; we say that y is the *successor* of x, noted $succ_>(x) = y$, if and only if $y > x$ and there is no $z \in S$ such that $y > z > x$. Similarly, we say that y is the *predecessor* of x, noted $pre_>(x) = y$, if and only if $x > y$ and there is no $z \in S$ such that $x > z > y$. Obviously enough, it can be the case that an element of a scale has no successor and/or no predecessor (just consider the scale $\mathbb{R} \cup \{-\infty, +\infty\}$ and any element of it).

Given a pair i from Int_S, one can associate to i the normal representation $i\downarrow \in Int_S$ of the corresponding interval. $i\downarrow$ is defined as $[]$ if the corresponding interval is empty; otherwise, $i\downarrow$ is defined as follows:

$$\begin{array}{lcll} l(i\downarrow) & = & l(i) & \text{if } lc(i) \text{ or} \\ & & & l(i) \text{ does not have a successor} \\ & = & succ_>(l(i)) & \text{otherwise} \end{array}$$

$$\begin{array}{lcll} lc(i\downarrow) & = & lc(i) & \text{if } lc(i) \text{ or} \\ & & & l(i) \text{ does not have a successor} \\ & = & true & \text{otherwise} \end{array}$$

$$\begin{array}{lcll} u(i\downarrow) & = & u(i) & \text{if } rc(i) \text{ or} \\ & & & u(i) \text{ does not have a predecessor} \\ & = & pre_>(u(i)) & \text{otherwise} \end{array}$$

$$\begin{array}{lcll} rc(i\downarrow) & = & rc(i) & \text{if } rc(i) \text{ or} \\ & & & u(i) \text{ does not have a successor} \\ & = & true & \text{otherwise} \end{array}$$

Considering the previous example again, where S is the set of integers, the pair $i = (2, 6)$ denoting the subset $\{3, 4, 5\}$ of S is such that $i \downarrow = [3, 5]$. Indeed, 3 is the successor of 2 and 5 is the predecessor of 6.

For many scales, normalizing representations of non-empty intervals is useless. Thus, it is easy to show that every pair representation of a non-empty interval over a *dense* scale is equal to its normal representation (a scale S is dense if and only if for every $x, y \in S$ such that $x < y$, thereq exists $z \in S$ such that $x < z < y$; every dense scale is infinite and no element of it has a successor or a predecessor). Dense scales have the interesting feature of always enabling "intermediate values".

In the following, without any loss of generality as to expressiveness, *we assume that every pair representation of an interval is its normal representation*. Especially, whenever we consider a pair representation i of an interval, i must be considered as a short for $i \downarrow$. We could easily ensure this normalization condition by considering dense scales, only, but this would restrict the setting unnecessarily (especially, some "natural" scales (e.g. when numbers represent amounts of money) are not dense.

$(Int_S, \subseteq)$ is a lattice, $\emptyset$ is the least element of Int_S and $S = [\bot, \top]$ is the greatest one. Let $>_{Int_S}$, $\sim_{Int_S}$ and $\geq_{Int_S}$ be the relations from $Int_S \times Int_S$ s.t. for every pair of non-empty intervals i_1 and i_2, we have:

- $i_1 >_{Int_S} i_2$ if and only if
  · $l(i_1) \geq u(i_2)$ if not $rc(i_2)$ and not $lc(i_1)$,
  · $l(i_1) > u(i_2)$ otherwise;
- $i_1 \sim_{Int_S} i_2$ if and only if $i_1 \cap i_2 \neq \emptyset$;
- $i_1 \geq_{Int_S} i_2$ if and only if $i_1 >_{Int_S} i_2$ or $i_1 \sim_{Int_S} i_2$.

Let S be a scale. Given an element s_1 of S, each relational operator $r \in \{=, \leq, \geq, <, >\}$ characterizes the interval $r(s_1) = \{s_2 \in S \mid s_2 \; r \; s_1\}$ of Int_S. Each set $r(s_1)$ consists of the set of all elements from S which are located between two elements of S among $s_1, \top, \bot$, one of them being s_1:

- $= (s_1) = [s_1, s_1] = \{s_1\}$;
- $\leq (s_1) = [\bot, s_1]$;
- $\geq (s_1) = [s_1, \top]$;
- $< (s_1) = [\bot, s_1)$;
- $> (s_1) = (s_1, \top]$.

2.3 Computational complexity

One of the main purposes of complexity theory (see e.g., [21] for details) is to classify problems according to their worst case requirements on computational resources depending on the size of the input. In this framework, a problem is a generic question, i.e., a set of specific instances. Specifically, a decision problem is one that has only "yes" and "no" as possible answers. Formally, a decision problem can be considered as a formal language consisting of the set of all its "yes" instances.

It is usually acknowledged that a practicable algorithm is one that runs in time polynomial in the size of the input (assuming a standard model of computation, e.g., a sequential deterministic Turing machine). Accordingly, a decision problem is considered tractable if and only if there exists an algorithm that can classify every instance of it in a number of computational steps that is polynomial in the size of the instance.

The class of all decision problems that are solvable in polynomial time is denoted by $\mathsf{P(TIME)}$. All the remaining ones require exponential time (in practice) and are considered intractable. In order to determine whether a problem is tractable or not, it is sufficient to point out an efficient algorithm to solve it, or to prove that such an algorithm cannot exist.

Unfortunately, for many problems, no polynomial time algorithms are known but today, nobody knows how to prove that super-polynomial time is actually required. For all these problems, the frontier between tractable problems and intractable ones is not fine-grained enough to classify them in a computationally valuable way: there is a need for more refined classes for problems for which one does not know whether they belong to P.

The notion of non-deterministic Turing machine is an important tool to achieve this goal. Thus, the class of all languages (encoding decision problems) that can be recognized in polynomial time by such a non-deterministic machine is denoted by NP. Because a deterministic Turing machine can be considered as a non-deterministic one, the inclusion $\mathsf{P} \subseteq \mathsf{NP}$ is established; however, the converse is the famous open problem: $\mathsf{P} \overset{?}{=} \mathsf{NP}$ (that is conjectured false).

Among all the problems in NP, the hardest ones are those from which every problem in NP can be *polynomially many-one reduced*: such problems are referred to as NP-*complete*. If any of them has a polynomial algorithm, then $\mathsf{P} = \mathsf{NP}$ holds. Accordingly, it is believed that it is impossible to solve NP-complete problems in (deterministic) polynomial time. SAT, the problem of determining whether a propositional formula in conjunctive normal form is satisfiable, is the prototypical NP-complete problem. Its complementary problem UNSAT (determining whether a propositional formula in conjunctive normal form is unsatisfiable) is not necessarily in NP (in contrast to P, NP is not known to be closed under complementation). It is assigned to the class coNP that contains the complementary problems to problems of NP. It is conjectured that $\mathsf{NP} \neq \mathsf{coNP}$.

To go further into the classification of non-efficiently solvable problems, another important tool is the notion of Turing machine (deterministic or non-deterministic) *with oracle*, i.e., a Turing machine with a black box which is able to decide the membership to some languages in a single operation. Let X be a class of decision problems (i.e., languages). P^X (resp. NP^X) is the class of all decision problems that can be solved in polynomial time using a deterministic (resp. non-deterministic) Turing machine that can use an oracle for deciding the membership to any language from X for “free” (i.e., within a constant, unit time). For instance P^{NP} is the class of problems solvable in polynomial time by a deterministic Turing machine with an oracle for any language in NP.

On this basis, the classes Δ_k^p, Σ_k^p and Π_k^p are defined by:

- $\Delta_0^p = \Sigma_0^p = \Pi_0^p = \mathsf{P}$,
- $\Delta_{k+1}^p = \mathsf{P}^{\Sigma_k^p}$,
- $\Sigma_{k+1}^p = \mathsf{NP}^{\Sigma_k^p}$,
- $\Pi_{k+1}^p = \mathsf{co}\Sigma_{k+1}^p = \mathsf{coNP}^{\Sigma_k^p}$,

Thus, $\Delta_1^p = \mathsf{P}$, $\Sigma_1^p = \mathsf{NP}$, $\Pi_1^p = \mathsf{coNP}$ and $\Delta_2^p = \mathsf{P}^{\mathsf{NP}}$. $\Delta_2^p[\mathcal{O}(log\ n)]$ (also referred to as Θ_2^p) is the class of problems which can be decided in polynomial time using only logarithmically many calls to an NP oracle.

The polynomial hierarchy PH is the union of all Σ_k^p (for k non-negative integer). A decision problem is said to be at the k^{th} level of PH if and only if it belongs to Δ_{k+1}^p, and is either Σ_k^p-hard or Π_k^p-hard. While it is easy to check that both $\Delta_k^p \subseteq \Sigma_{k+1}^p$, $\Delta_k^p \subseteq \Pi_{k+1}^p$, $\Sigma_k^p \subseteq \Delta_{k+1}^p$, and $\Pi_k^p \subseteq \Delta_{k+1}^p$ hold for every k, it is unknown whether the inclusions are proper or not (but it is conjectured that it is the case). Thus, it is strongly believed that the polynomial hierarchy does not collapse, i.e., is a truly infinite hierarchy (for every integer k, $\mathsf{PH} \neq \Sigma_k^p$).

All the problems in PH can be solved in (simple) exponential time on deterministic Turing machines. Nevertheless, the level where a given decision problem lies in the polynomial hierarchy can be viewed as a measure of its complexity, since this level intuitively corresponds to the number of independent sources of intractability to be dealt with. Especially, removing one source of intractability by focusing on a restriction where it disappears is not enough to question the other sources of intractability. Thus the higher a problem in the polynomial hierarchy the more complex, in the sense that more severe restrictions have to be applied to get a tractable case.

3 Representing Interval Orders

3.1 Interval orders

Let us first give a formal definition of interval orders:

Definition 3 *An* interval order $R \subseteq A \times A$ *is a relation which is:*

- *Complete:* $\forall a_1, a_2 \in A$, $(a_1, a_2) \in R$ *or* $(a_2, a_1) \in R$.
- *Ferrers:* $\forall a_1, a_2, a_3, a_4 \in A$, *if* $(a_1, a_2) \in R$ *and* $(a_3, a_4) \in R$, *then* $(a_1, a_4) \in R$ *or* $(a_3, a_2) \in R$.

Because they are complete, interval orders are reflexive relations. However, they are not necessarily transitive relations (and this is where they depart from preorders). Nevertheless, the strict preference relation P_R associated to an interval order R is transitive and Ferrers.

Example 2 *Let* $A = \{a, b, c, d\}$ *and let* $R = \{(a, a), (a, b), (a, c), (a, d), (b, a), (b, b), (b, c), (b, d), (c, b), (c, c), (c, d), (d, c), (d, d)\}$. *$R$ is an interval order over A (but not a preorder over A since e.g., we have $(c, b) \in R$ and $(b, a) \in R$ but $(c, a) \notin R$). We have* $I_R = \{(a, a), (a, b), (b, a), (b, b), (b, c), (c, b), (c, c), (c, d), (d, c), (d, d)\}$ *and* $P_R = \{(a, c), (a, d), (b, d)\}$.

The following representation theorem due to Fishburn [10] shows that any interval order over a countable set can be characterized by intervals over $\mathbb{R}$:

Proposition 1 *[10] Let A be a countable set. A binary relation $R \subseteq A \times A$ is an interval order if and only if there exist a mapping g from A to $\mathbb{R}$ and a mapping q from $\mathbb{R}$ to $\mathbb{R}^+$*[5] *such that for any $a_1, a_2 \in A$, we have*

$$(a_1, a_2) \in R \textit{ if and only if } g(a_1) + q(g(a_1)) \geq g(a_2).$$

Indeed, the mappings g and q come down to associating to any element a of A an interval $[l_a = g(a), u_a = g(a) + q(g(a))]$ over $\mathbb{R}$ such that for any $a_1, a_2 \in A$, we have:

- $(a_1, a_2) \in P_R$ if and only if $l_{a_1} > u_{a_2}$,
- $(a_1, a_2) \in I_R$ if and only if $l_{a_1} \leq u_{a_2}$ and $l_{a_2} \leq u_{a_1}$.

Example 3 *Let g and q such that: $g(a) = 6$, $g(b) = 4$, $g(c) = 2$, $g(d) = 0$, and $q(x) = 3$ for every $x \in \mathbb{R}$. (g, q) represents the relation R given in the previous example. An interval representation of R may be given as a set of pairs where the*

[5] $\mathbb{R}^+$ is the set of non-negative real numbers.

first element of each (ordered) pair is an alternative and the second one is the interval associated to this alternative: $\{(a, [6, 9]), (b, [4, 7]), (c, [2, 5]), (d, [0, 3])\}$ *is an interval representation of* R*. The following figure illustrates this interval representation.*

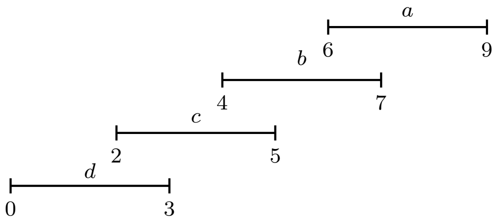

Fig. 1. Interval representation of the example 3

When A is finite, Fishburn's theorem (Proposition 1) can be specialized so that closed intervals of $\mathbb{N}$ included in $[0, 2 \times Card(A) - 1]$ are enough. Indeed, let R be an interval order defined on the finite set $A = \{a_1, a_2, \ldots, a_n\}$ of alternatives; taking advantage of Proposition 1, let l_{a_i} (resp. u_{a_i}) be the lower (resp. upper) bound of the real interval associated to the alternative a_i. We consider now the set of endpoints $E = \{l_{a_i}, u_{a_i} \mid a_i \in A\}$ naturally ordered and we denote by rank(l_{a_i}) (resp. rank(u_{a_i})) the number of elements of E which are strictly lower than l_{a_i} (resp. u_{a_i}) (for instance rank$(l_{a_i}) = 0$ if and only if $\forall a_j \in A,\ l_{a_i} \leq l_{a_j}$). Then we have for all $a_i, a_j \in A$:

- $(a_i, a_j) \in P_R$ if and only if rank$(l_{a_i}) >$ rank(u_{a_j}),
- $(a_i, a_j) \in I_R$ if and only if rank$(l_{a_i}) \leq$ rank(u_{a_j}) and rank$(l_{a_j}) \leq$ rank(u_{a_i}).

3.2 Weighted bases

We are now ready to define the notion of weighted base, at the syntax level first and then from a semantical point of view:

Definition 4 *Let* PS *be a finite set of atomic propositions and let* S *be a scale. A* weighted base B *over* PS *and* S *is a finite multiset* B *of 5-tuples* $(\phi, r_1, s_1, r_2, s_2)$ *where* ϕ *is a formula of* $PROP_{PS}$*,* r_1 *(resp.* r_2*) is a relational operator from* $\{=, >, \geq\}$ *(resp.* $\{=, <, \leq\}$*) and* s_1, s_2 *are elements of* S*.*

At the semantic level, the 4-tuple (r_1, s_1, r_2, s_2) represents the interval associated to the formula ϕ. As in the following example, generally s_1 is less than or equal to s_2, in the contrary case we have an empty interval (for more details please see Subsection 2.2).

Example 4 *Let* $PS = \{x, y, z\}$ *and* $S = [\bot = 0, \top = 100] \subseteq \mathbb{R}$ *naturally ordered.* $B = \{(\neg x, \geq, 30, \leq, 80), (\neg z, \geq, 20, \leq, 60), (x \wedge \neg y, \geq, 20, \leq, 40), (x \wedge z, \geq, 50, \leq, 100)\}$ *is a weighted base over* PS *and* S*.*

When B is a weighted base, $Card(B)$ is the number of 5-tuples B contains. $Var(B)$ denotes the set of all atoms from PS occurring in B, i.e., $Var(B) = \bigcup_{(\phi,r_1,s_1,r_2,s_2)\in B} Var(\phi)$. Finally, the size $|B|$ of B is equal to the sum of the sizes of each 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ in it, where the size of $(\phi, r_1, s_1, r_2, s_2)$ is equal to $|\phi|+|s_1|+|s_2|+2$; if S is a numerical scale, then its elements are supposed to be represented in binary notation.

Intuitively, 5-tuples $(\phi, r_1, s_1, r_2, s_2)$ must be viewed as pieces of preferential evidence, corresponding to imprecise evaluations of the alternatives under consideration w.r.t. different criteria or w.r.t. different sources. Thus, on the example above, $(\neg x, \geq, 30, \leq, 80)$ means that for a given criterion or source, the utility of any model of $\neg x$ is between (or equal to) 30 and 80.

Remark 1 *In order to alleviate the notations, when one of the endpoint s_1 or s_2 of the interval associated to a formula is also one of the bounds $\bot$, $\top$ of the scale S, a triple (ϕ, r, s) may be used instead of $(\phi, r_1, s_1, r_2, s_2)$. For example the last 5-tuple of the previous example may be written as $(x \wedge z, \geq, 50)$ since 100 is the greatest element of $S = [0, 100]$.*

Putting the criteria or sources together calls for aggregation functions:

Definition 5 *Let f be any mapping from $Int_S \times Int_S$ to Int_S which is associative, commutative and has a neutral element $n_f \in Int_S$. f is called an S-aggregation function. It is extended to a mapping from any finite subset of Int_S to Int_S, also referred to as f, and defined inductively by:*

- $f(\emptyset) = n_f$,
- $f(\{i\}) = i$,
- $f(\{i_1, i_2, \ldots, i_n\}) = f(i_1, f(\{i_2, \ldots, i_n\}))$.

We are now ready to define the notion of satisfaction of a weighted base by an interpretation. Each interpretation can be viewed as an imprecise, yet qualitative utility function on S. $I(\omega)$ is the interval where the actual (qualitative) utility of world ω lies. The notion of satisfaction is defined in a compositional way and depends on the chosen S-aggregation function f:

Definition 6 *Let PS be a finite set of atomic propositions and let S be a scale. Let I be an interpretation over PS and S, i.e., a mapping from 2^{PS} to Int_S. Let f be an S-aggregation function. I* f-satisfies *a weighted base B over PS and S, noted $I \models_f B$ if and only if $\forall \omega \in 2^{PS}$, we have*

$$I(\omega) \subseteq f(\{r_1(s_1) \cap r_2(s_2) \mid (\phi, r_1, s_1, r_2, s_2) \in B \text{ and } \omega \models \phi\}).$$

The neutral element of the S-aggregation function will be used as the interpretation of a world which satisfies none of the formulae of the given weighted base.

Among the possible S-aggregation functions f for any scale S are $\cap$, whose neutral element $n_\cap$ is $S = [\bot, \top]$ and $+$ which is such that for any $i_1, i_2 \in Int_S$, $i_1 + i_2$ is the smallest interval (w.r.t. $\subseteq$) containing both i_1 and i_2. Clearly enough, the neutral element n_+ is the empty interval $\emptyset$, noted $[]$. It is obvious to show that $\cap$ is the infimum function for the lattice $(Int_S, \subseteq)$, and that $+$ is the supremum function for this lattice.

Simple S-aggregation functions are min and max: when none of i_1 and i_2 is the empty interval, $l(i_1\ \mathit{min}\ i_2)$ is the minimum of $l(i_1)$ and $l(i_2)$, $u(i_1\ \mathit{min}\ i_2)$ is the minimum of $u(i_1)$ and $u(i_2)$,

$$\begin{array}{rcll}
lc(i_1\ \mathit{min}\ i_2) & = & lc(i_1) & \text{if } l(i_1) < l(i_2) \\
 & = & lc(i_2) & \text{if } l(i_2) < l(i_1) \\
 & = & lc(i_1) \text{ or } lc(i_2) & \text{otherwise} \\
rc(i_1\ \mathit{min}\ i_2) & = & rc(i_1) & \text{if } u(i_1) < u(i_2) \\
 & = & rc(i_2) & \text{if } u(i_2) < u(i_1) \\
 & = & rc(i_1) \text{ and } rc(i_2) & \text{otherwise}
\end{array}$$

In the remaining case, i.e., if at least one of i_1, i_2 is $[]$, then $i_1\ \mathit{min}\ i_2$ also is the empty interval (i.e., $[]$ is an absorbing element). We have $n_{min} = [\top, \top]$.

When none of i_1 and i_2 is the empty interval, $l(i_1\ \mathit{max}\ i_2)$ is the maximum of $l(i_1)$ and $l(i_2)$, $u(i_1\ \mathit{max}\ i_2)$ is the maximum of $u(i_1)$ and $u(i_2)$,

$$\begin{array}{rcll}
lc(i_1\ \mathit{max}\ i_2) & = & lc(i_1) & \text{if } l(i_1) > l(i_2) \\
 & = & lc(i_2) & \text{if } l(i_2) > l(i_1) \\
 & = & lc(i_1) \text{ and } lc(i_2) & \text{otherwise} \\
rc(i_1\ \mathit{min}\ i_2) & = & rc(i_1) & \text{if } u(i_1) > u(i_2) \\
 & = & rc(i_2) & \text{if } u(i_2) > u(i_1) \\
 & = & rc(i_1) \text{ or } rc(i_2) & \text{otherwise}
\end{array}$$

In the remaining case, i.e., if at least one of i_1, i_2 is $[]$, then $i_1\ \mathit{max}\ i_2$ also is the empty interval. We have $n_{max} = [\bot, \bot]$.

Another simple S-aggregation function for a numerical scale $S = [\bot, \top] \subseteq \mathbb{R} \cup \{-\infty, +\infty\}$ (such that S contains 0) is sum: when none of i_1 and i_2 is the empty interval $[]$, let ls be the sum of $l(i_1)$ and $l(i_2)$ and rs be the sum of $u(i_1)$ and $u(i_2)$; we have:

$$\begin{array}{lcll} l(i_1 \mathit{sum}\, i_2) & = & ls & \text{if } ls \in S \\ & = & \bot & \text{if } ls < \bot \\ & = & \top & \text{if } ls > \top \\ u(i_1\ \mathit{sum}\, i_2) & = & rs & \text{if } rs \in S \\ & = & \bot & \text{if } rs < \bot \\ & = & \top & \text{if } rs > \top \end{array}$$

$$\begin{array}{lcll} lc(i_1 \mathit{sum}\, i_2) & = & lc(i_1) \text{ and } lc(i_2) & \text{if } ls \in S \\ & = & true & \text{otherwise} \\ rc(i_1\ \mathit{sum}\, i_2) & = & rc(i_1) \text{ and } lc(r_2) & \text{if } rs \in S \\ & = & true & \text{otherwise} \end{array}$$

In the remaining case, i.e., if at least one of i_1, i_2 is [], then i_1 *sum* i_2 also is the empty interval. We have $n_{sum} = [0, 0]$.

The choice of the S-aggregation function may depend on different notions, such as the nature and properties of order to represent or the context of decision problem, etc. For example, "+" may be used in a case where we do not want to lose any information and where the accuracy is not very important, "$\cap$" may be useful when the precision of the information is crucial. "*max*" (resp. "*min*") may represent an optimist (resp. pessimist) view. "*sum*" may be necessary in the case where the addition of values has a reasonable meaning.

For instance, the weighted base $\{(sea \wedge summer, \geq, 20), (sea \wedge summer, \ \leq, 30), (\neg sea \wedge summer, \geq, 10), (\neg sea \wedge summer, \leq, 25), (sea, \geq, -10), (sea, \leq, -5)\}$ can be used to represent preferences concerning the period (during summer or not) and the location (near the sea or not) of future vacations; 5-tuples can be concerned here with two criteria: pleasure and cost (for the last two). sum, min and max are (among others) natural ways to aggregate intervals associated to 5-tuples in this case.

The weighted base $\{(electric{-}guitar, \geq, 20), (electric{-}guitar, \leq, 1000), (electric{-}guitar, \geq, 200), (electric{-}guitar, \leq, 500), (electric{-}guitar, \geq, 300), (electric{-}guitar, \leq, 700)\}$ collects evidence given by three friends about the price of a quite good, yet cheap electric guitar. Here $\cap$ is a natural aggregation function if the sources are all completely reliable (it leads to uncertainty reduction).

The presence of several criteria or several sources explains why a multiset representation of weighted bases is more convenient than a set-based one: it can be the case that two criteria (or sources) evaluate alternatives in the same way; while this has

2^{PS}	$I_B^{\cap}$	I_B^{+}	$I_B^{\max}$	$I_B^{\min}$	I_B^{sum}
000	$[30, 60]$	$[20, 80]$	$[30, 80]$	$[20, 60]$	$[50, 100]$
001	$[30, 80]$	$[30, 80]$	$[30, 80]$	$[30, 80]$	$[30, 80]$
010	$[30, 60]$	$[20, 80]$	$[30, 80]$	$[20, 60]$	$[50, 100]$
011	$[30, 80]$	$[30, 80]$	$[30, 80]$	$[30, 80]$	$[30, 80]$
100	$[20, 40]$	$[20, 60]$	$[20, 60]$	$[20, 40]$	$[40, 100]$
101	$[]$	$[20, 100]$	$[50, 100]$	$[20, 40]$	$[70, 100]$
110	$[20, 60]$	$[20, 60]$	$[20, 60]$	$[20, 60]$	$[20, 60]$
111	$[50, 100]$	$[50, 100]$	$[50, 100]$	$[50, 100]$	$[50, 100]$

Table 1
Canonical interpretations.

no impact when the aggregation function under consideration is idempotent (like $\cap$ and $+$), this is not the case more generally (just consider sum).

3.3 Canonical interpretations

In the general case, a weighted base can be associated to many interpretations that f-satisfy it. However, one of them plays a specific role: the f-canonical interpretation associated to B.

Definition 7 *Let PS be a finite set of atomic propositions and let S be a scale. Let f be any S-aggregation function. The f-*canonical interpretation *I_B^f associated to the weighted base B over PS and S and the aggregation function f is the interpretation over PS and S given by: for any $\omega \in 2^{PS}$,*

$$I_B^f(\omega) = f(\{r_1(s_1) \cap r_2(s_2) \mid (\phi, r_1, s_1, r_2, s_2) \in B \textit{ and } \omega \models \phi\}).$$

Example 5 *Let $PS = \{x, y, z\}$ ordered in this way. Let S be the scale $[0, 100] \subseteq \mathbb{R}$ naturally ordered. $B = \{(\neg x, \geq, 30, \leq, 80), (\neg z, \geq, 20, \leq, 60), (x \wedge \neg y, \geq, 20, \leq, 40), (x \wedge z, \geq, 50, \leq, 100)\}$ is a weighted base over PS and S. Table 1 gives interval representations of $I_B^{\cap}$, I_B^{+}, $I_B^{\max}$, $I_B^{\min}$, and I_B^{sum}.*

I_B^f can be characterized as the least specific interpretation f-satisfying B, in the sense that for each interpretation I f-satisfying B and for each $\omega \in 2^{PS}$, we have $I(\omega) \subseteq I_B^f(\omega)$.

Note that the $\cap$-canonical interpretation associated to a weighted base is the least specific interpretation which satisfies every element of B.

Now, a notion of *f-equivalence* between weighted bases over PS and S can be easily defined: two bases B_1 and B_2 are said to be f-equivalent when the set of all interpretations f-satisfying B_1 is equal to the set of all interpretations f-satisfying B_2. It is easy to prove that two weighted bases are f-equivalent whenever they are associated to the same f-canonical interpretation:

Proposition 2 *Let PS be a finite set of atomic propositions and let S be a scale. Let B_1, B_2 be two weighted bases over PS and S. Let f be any S-aggregation function. B_1 and B_2 are f-equivalent if and only if $I^f_{B_1} = I^f_{B_2}$.*

Proof: Obvious.

□

In order to make precise the connection between I^f_B and the corresponding interval order, we still need two notions, namely the notion of f-consistent weighted base and the notion of f-possible world.

Obviously enough, the trivial interpretation $I_\emptyset$ s.t. $\forall\omega \in 2^{PS}$, $I(\omega) = []$ f-satisfies any weighted base B. For this reason, the notion of f-consistency of a weighted base cannot be defined as the existence of an interpretation f-satisfying it but has to be a bit more elaborate:

Definition 8 *Let PS be a finite set of atomic propositions and let S be a scale. Let f be any S-aggregation function. A weighted base B over PS and S is said to be* f-consistent *if and only if there exists an interpretation I over PS and S such that $I \models_f B$ and $I \neq I_\emptyset$.*

Example 6 *The base B given in Example 5 is f-consistent for f among $\cap$, $+$,* $\max$, $\min$ *and* sum.*The base* $\{(x, >, 3), (x, =, 2), (\neg x, <, 1), (\neg x, >, 3)\}$ *is not $\cap$-consistent but is $+$-consistent.*

It is easy to see that a weighted base B is $+$-consistent if and only if B contains at least one 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ such that $r_1(s_1) \cap r_2(s_2) \neq []$. Every weighted base B is $\min$-consistent, $\max$-consistent and sum-consistent except when it contains a 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ such that $r_1(s_1) \cap r_2(s_2) = []$ (i.e., the interval specified by r_1, s_1, r_2, s_2 is empty).

Now, the preimage of $\emptyset$ by I^f_B is a subset of 2^{PS} of worlds which are considered as impossible given B:

Definition 9 *Let PS be a finite set of atomic propositions and let S be a scale. Let B be a weighted base over PS and S. Let f be any S-aggregation function. Let ω be a world from 2^{PS}. ω is said to be* f-impossible *given B if and only if $I^f_B(\omega) = []$. ω is said to be* f-possible *when it is not f-impossible.*

$Possible_f(B)$ denotes the subset of 2^{PS} consisting of all worlds which are f-possible given B. Clearly enough, we have $Possible_f(B) = \emptyset$ if and only if B is not f-consistent if and only if $I_B^f = I_\emptyset$.

Example 7 *Let us consider again the base B used in Example 5. We have $Possible_\cap(B) = 2^{PS} \setminus \{101\}$.*

f-possible worlds correspond to feasible alternatives and f-consistency ensures that at least some alternative is feasible.

For instance, if we step back to the example of a person who wants to know whether she must buy or not an electric guitar, the case $I_B^\cap(electric - guitar) = []$ means that the pieces of information given by her three friends are jointly contradictory, hence we could not consider this world.

We are now ready to define the preference relations induced by a weighted base and an aggregation function:

Definition 10 *Let PS be a finite set of atomic propositions and let S be a scale. Let f be any S-aggregation function. Let B be a weighted base over PS and S. We define the binary relations $\succ_B^f$, $\sim_B^f$ and $\succeq_B^f \subseteq Possible_f(B) \times Possible_f(B)$ by: let $\omega_1, \omega_2 \in Possible_f(B)$,*

- *$\omega_1 \succ_B^f \omega_2$ if and only if $I_B^f(\omega_1) >_{Ints_S} I_B^f(\omega_2)$;*
- *$\omega_1 \sim_B^f \omega_2$ if and only if $I_B^f(\omega_1) \sim_{Ints_S} I_B^f(\omega_2)$;*
- *$\omega_1 \succeq_B^f \omega_2$ if and only if $\omega_1 \succ_B^f \omega_2$ or $\omega_1 \sim_B^f \omega_2$.*

The next result is a soundness one: it shows that every weighted base and S-aggregation function actually specifies an interval order over S:

Proposition 3 *Let PS be a finite set of atomic propositions and let S be a scale. Let f be any S-aggregation function. Let B be a weighted base over PS and S. $\succeq_B^f$ is an interval order on $Possible_f(B)$. $\sim_B^f$ (resp. $\succ_B^f$) is its symmetric (resp. asymmetric) part.*

Proof: The result follows directly from Proposition 1.

□

It is obvious that depending on the chosen S-aggregation function the resulting interval order R may differ. For instance, in Example 5, the use of five different S-aggregation functions leave place to four different interval orders (see Table 1): all the worlds are indifferent when "+" and "$\max$" are used (so we obtain a preorder); with "$\cap$", all the relations, except the one between 111 and 100 ($111\ P_R\ 100$), are indifference and with "$\min$", all the relations, except $111\ P_R\ 100$ and $111\ P_R\ 101$, are indifference and finally with "sum", all the relations, except $101\ P_R\ 110$, are indifference. It is easy to remark that in all of the last four cases the indifference relation associated to the interval order is not transitive.

Finally, it is easy to see that two f-equivalent weighted bases specify the same interval order but the converse is false (i.e., an order may be represented by two different, yet not f-equivalent bases); for instance consider a numerical scale like $\mathbb{R} \cup \{-\infty, +\infty\}$; consider two weighted bases B_1 and B_2 and imagine the case where B_2 is defined like B_1 except that all the values are multiplied by 2. For that reason we define another, less demanding, notion of equivalence that we call (f, g)-*order-equivalence*: if f is an S_1-aggregation function and g is an S_2-aggregation function, then a weighted base B_1 over PS and S_1 is said to be (f, g)-order-equivalent to a weighted base B_2 over PS and S_2 if and only if $\succeq^f_{B_1} = \succeq^g_{B_2}$.

3.4 Simplicity and modularity

Basically, in our approach, the connection between the weighted base representation B of an interval order and the order itself is as follows: 5-tuples $(\phi, r_1, s_1, r_2, s_2)$ of B are viewed as pieces of preferential evidence and are aggregated using a suitable function f; the corresponding f-canonical interpretation maps each interpretation (representing an alternative) to an interval; the associated interval order follows in an obvious way once alternatives associated to the empty interval have been removed.

In our opinion, this semantics is simple; of course, simplicity has no formal definition but let us stress that the semantics of weighted propositional formulae (or penalty-based representations) as considered in [16,5,17,4] and used for representing cardinal preferences is very close to the semantics we point out. Indeed, in penalty logic, weighted bases P are multisets of pairs of the form (ϕ, α) where $\phi \in PROP_{PS}$ and $\alpha \in \mathbb{R} \cup \{-\infty, +\infty\}$; α is the penalty to be paid whenever ϕ is not satisfied; associated to each weighted base P is a propositional formula K used to discriminate impossible worlds: the possible worlds are restricted to the models of K. Disutilities α are aggregated in an additive way so that the disutility of any world $\omega \in Mod(K)$ is given by $d(\omega) = \sum_{(\phi,\alpha)\in P \ | \ \omega \not\models \phi} \alpha$.

The main differences between penalty-based representations of cardinal preferences and our framework are as follows: in penalty-based representations, the purpose is to represent complete preorders (and not interval orders); each piece of preferential evidence allows to map worlds (representing alternatives) to values from a numerical scale S (disutilities) and not to intervals over any S; sum is the aggregation function (while we consider other functions in our framework); finally, a further propositional formula K is used to discriminate impossible worlds in the penalty-based approach (while we consider as impossible those worlds associated to the empty interval).

The modularity of the representation in our framework is achieved, at the syntax level, by the multiset representation of weighted base (5-tuples can be considered in a separate way) and at the semantical level, by the compositionality of the construction of the f-canonical interpretation.

4 Expressiveness and Spatial Efficiency

After the soundness result given by Proposition 3, let us first show that a completeness result can be achieved for some specific choices of scales S, whatever the chosen S-aggregation function f. Of specific interest is the scale $S = [0, 2 \times Card(A) - 1] \subseteq \mathbb{N}$.[6] From Fishburn's result, it comes that any interval order over a finite set of alternatives A can be represented by a weighted base in our setting:

Proposition 4 *Let A be a finite set of alternatives and $\succeq$ an interval order on A. For every S-aggregation function f with $S = [0, 2 \times Card(A) - 1] \subseteq \mathbb{N}$, there exists a set of propositional atoms PS and a weighted base B over PS and S such that there is a bijection b between A and $Possible_f(B)$ and for every a_1, $a_2 \in A$, we have $a_1 \succeq a_2$ if and only if $b(a_1) \succeq^f_B b(a_2)$.*

Proof: Using Proposition 1, we associate an interval $[l_a, u_a]$ to each alternative a of the set A. We define a set of propositional atoms PS such as $Card(A) \leq 2^{Card(PS)}$, and a weighted base B as follows. We associate a world $\omega_a \in 2^{PS}$ to each alternative a in a bijective way (i.e., $b(a) = \omega_a$). For each ω_a we consider a formula ϕ_a satisfied only by ω_a (this is trivially possible), and add to B (initially empty) the constraint $(\phi_a, \geq, l_a, \leq, u_a)$. It is easy to see that for every $a_1, a_2 \in A$, we have $a_1 \succeq a_2$ if and only if $b(a_1) \succeq^f_B b(a_2)$.

□

Obviously enough, this proposition can be trivially extended to every case a scale isomorphic to $[0, 2 \times Card(A) - 1] \subseteq \mathbb{N}$ (naturally ordered) is considered. This proposition also shows that the choice of the aggregation function has no impact on the expressiveness issue, as soon as such a scale is considered.

Contrastingly, imposing strong restrictions on the scale (whatever the chosen set of atoms and the aggregation function) easily leads to losing full expressiveness; for instance, if the scale under consideration is degenerate so that it contains only the two extremal elements $\bot$ and $\top$, then whatever the choice of PS and f, the corresponding languages of weighted bases cannot achieve the representation of any interval order R for which a P_R-chain with more than two elements exists.

[6] Alternatively, we could consider the scale $S = \mathbb{R} \cup \{-\infty, +\infty\}$ where $\leq$ extends the standard ordering on real numbers so that $\bot = -\infty$ is the least element of S and $\top = +\infty$ is the greatest element of S. Our preference to integers instead of reals is motivated by representational and computational issues.

Similarly, $Card(B)$ has a strong impact on the expressiveness issue: the interval representation of the order given by B contains at most $2^{Card(B)}$ elements, hence an interval order R for which a P_R-chain with more than $2^{Card(B)}$ elements exists cannot be represented by B, whatever the choices of S and f.

Let us now show that a language of weighted bases for representing interval orders is strictly more expressive than the language of penalty-based representations for cardinal preferences:

Proposition 5 *Every complete preorder on a set of alternatives can be represented in the language of weighted bases using $S = \mathbb{R} \cup \{-\infty, +\infty\}$ as the scale and sum as the S-aggregation function. Contrastingly, it is not the case that every interval order can be represented in the language of penalty logic.*

Proof: It is enough to prove the first point since it is known that the language of penalty logic can express complete preorders but nothing more general (see [5]). We do it by pointing out a translation from the language of penalty-based representations of complete preorders to the language of weighted bases. Let $P = \{(\phi_i, \alpha_i) \mid i \in I\}$ be a penalty base and let K be the associated constraint (a propositional formula) such that the feasible alternatives correspond to the models of K (i.e., the possible worlds are restricted to the models of K). We associate to P and K in linear time the weighted base

$$B_{P,K} = \{(\neg\phi_i, =, \alpha_i) \mid i \in I\} \cup \{(\neg K, <, 0), (\neg K, >, 0)\}.$$

Clearly, we have $I^{sum}_{B_{P,K}}(\omega) = []$ for every $\omega \not\models K$, due to the 3-tuples $(\neg K, <, 0)$, $(\neg K, >, 0)$ in $B_{P,K}$; furthermore $I^{sum}_{B_{P,K}}(\omega) = [d(\omega), d(\omega)]$ for every $\omega \in Mod(K)$ where $d(\omega)$ represents the disutility of the world ω ($d(\omega) = \sum_{(\neg\phi_i, =, \alpha_i) \in B_{P,K} \mid \omega \models \neg\phi_i} \alpha_i$ $= \sum_{(\phi_i, \alpha_i) \in P \mid \omega \not\models \phi_i} \alpha_i$). As a consequence, for any $\omega_1, \omega_2 \in Mod(K)$, we have $d(\omega_1) > d(\omega_2)$ if and only if $\omega_1 \succ^{sum}_{B_{P,K}} \omega_2$, and $d(\omega_1) = d(\omega_2)$ if and only if $\omega_1 \sim^{sum}_{B_{P,K}} \omega_2$. This shows the result of the translation as a faithful representation of the complete preorder induced by d.

□

Let us finally turn to the spatial efficiency issue and briefly compare our representation languages using weighted bases to both the explicit representation and the interval representations or interval orders.

The explicit representation of interval orders given by Definition 3 (as sets of ordered pairs of alternatives from a finite set A) is just as compact as its representation using intervals over $\mathbb{R}$, i.e., as sets S_I of pairs (a, i) where a is an alternative and i an interval. Indeed, from Fishburn's theorem (see Proposition 1), we immediately get that for any pair (a_1, a_2) of alternatives occurring in S_I, $(a_1, a_2) \in R$ if and only if $l_{a_1} > u_{a_2}$ or ($l_{a_1} \leq u_{a_2}$ and $l_{a_2} \leq u_{a_1}$). Hence from any set S_I of pairs (a, i) representing an interval order R, deriving the corresponding explicit representation of R requires $\mathcal{O}(|S_I|^2)$ time in the worst case (hence a polynomial amount of space).

Conversely, the explicit representation of an interval order being given, an interval representation of this order can be computed in polynomial time. For instance Doignon [8] pointed out a polynomial time algorithm for computing the minimal interval representation of an interval order R over a set A of alternatives; here a representation of R is a pair (l, u) of mappings from A to $Int_{\mathbb{R}^+}$; a representation (l^*, u^*) of R is minimal if and only if for all the representations (l, u) of R and for all a in A, $l^*(a) \leq l(a)$ and $u^*(a) \leq u(a)$.[7] . In the same vein Isaak [13] gave a polynomial time algorithm for computing an interval representation of an interval order, which minimizes the number of different left and right bounds of intervals.

Contrastingly, the ability offered by propositional logic to represent sets of worlds in a compact way is enough for ensuring that the languages of weighted bases (at least those which enable the representation of every interval order) are strictly more compact than both the explicit representation and the interval representations. Indeed, the problem with the explicit representation and the interval representations is that they both require an explicit representation of the alternatives. Assume that $A = 2^{PS}$ for a given set PS of n symbols: the explicit representation of any interval order R over A contains at least 2^n elements; this is also the case for any interval representation of A. However, some interval orders over A can be represented in a much more compact way using a weighted base. For instance, if $PS = \{x_1, \ldots, x_n\}$, $B = \{(x_1 \vee \ldots \vee x_n, =, \top)\}$ can be used for representing a complete preorder where all the models of $x_1 \vee \ldots \vee x_n$ are indifferent but strictly preferred to the remaining world (where all atoms are set to $false$) (it is enough to consider max as the aggregation function).

5 Complexity Results

In this section, we investigate the complexity of exploiting weighted bases as interval order representations. We aim at deriving general results, in the sense that we do not focus on specific scales S and aggregation functions f. S and f are parameters of the decision problems we consider (especially, S – which can be infinite – is not part of the input).

Nevertheless, we need to put some requirements on the acceptable S and f. We first assume that a normal representation of any interval $i \in Int_S$ can be computed in time polynomial in the size of i(please see Page 7 for the presentation of a normal representation). Especially, deciding whether a pair representation denotes the empty interval can be achieved in polynomial time. This assumption is not very demanding (dense scales satisfy it and finite scales as well since they are isomorphic to a closed interval of the set $\mathbb{N}$of non-negative integers). We also assume that f

[7] In this study, each endpoint of $l^*(a)$ and $u^*(a)$ with $a \in A$ are defined as integer multiples of a positive real number. As a consequence the ”true scale” is a discrete one.

is a linear time function (i.e., for any x and y in the domain of f, $f(x, y)$ can be computed in time linear in the size of x plus the size of y). Under this assumption (which holds for any of $\cap$, $+$, min, max and sum), it is obvious that for any world $\omega \in 2^{PS}$ and any weighted base B, the value of $I_B^f(\omega)$ (as a normal interval) can be computed in time polynomial in the size of B plus the size of ω. We finally also assume that the ordering $\leq$ on any scale S can be decided in polynomial time.

The first question to be addressed when taking advantage of a weighted base B to represent a non-trivial interval order is whether B is f-consistent.

Definition 11 f-CONSISTENCY *is the decision problem given by*

- **Input:** *A weighted base B over PS and S.*
- **Question:** *Is B f-consistent?*

Now, since the set of alternatives characterized by a weighted base is not always equal to the set of all worlds, another important question is to determine when a world is among the alternatives:

Definition 12 f-POSSIBILITY *is the decision problem given by*

- **Input:** *A weighted base B over PS and S and a world $\omega \in 2^{PS}$.*
- **Question:** *Is ω f-possible given B?*

Since many different bases share the same f-canonical interpretation in the general case, it is important to identify the complexity of deciding whether two given bases are f-equivalent; indeed, if two bases are f-equivalent then they represent the same interval order:

Definition 13 f-EQUIVALENCE *is the decision problem given by*

- **Input:** *Two weighted bases B_1 and B_2 over PS and S.*
- **Question:** *Are B_1 and B_2 f-equivalent?*

Similarly, it is important to identify the complexity of deciding whether a first base is (f, g)-order-equivalent to a second base (since, as we have seen, it can be the case that two bases that are not f-equivalent nevertheless represent the same interval order); we assume here that both f and g are linear time aggregation functions (possibly over different scales S_1 and S_2):

Definition 14 (f, g)-ORDER-EQUIVALENCE *is the decision problem given by*

- **Input:** *A weighted base B_1 over PS and S_1, and a weighted base B_2 over PS and S_2.*
- **Question:** *Is B_1 (f, g)-order-equivalent to B_2?*

The following three decision problems are in some sense natural problems when dealing with compactly represented interval orders (and, as such, have been considered in other papers from the literature dealing with compact representation of preferences, see [17]); let op be an element of $\{\succeq, \succ, \sim\}$:

Definition 15 *f-COMPARISON(op) is the decision problem given by*

- **Input:** *A weighted base B over PS and S and two worlds $\omega_1, \omega_2 \in Possible_f(B)$.*
- **Question:** *Is $\omega_1 op^f_B \omega_2$ true?*

Definition 16 *f-NON-DOMINANCE is the decision problem given by:*

- **Input:** *A weighted base B over PS and S and a world $\omega \in Possible_f(B)$.*
- **Question:** *Is ω undominated w.r.t. $\succ^f_B$ in the set of all worlds f-possible given B?*

Definition 17 *f-CAND-OPT-SAT is the decision problem given by:*

- **Input:** *A weighted base B over PS and S and a formula $\phi \in PROP_{PS}$.*
- **Question:** *Does there exist an undominated world ω w.r.t. $\succ^f_B$ in the set of all worlds f-possible given B s.t. $\omega \models \phi$?*

We have derived the following results:

Theorem 1 *f-CONSISTENCY is in NP; it is NP-complete for $f = \cap$ and in P for $f = +$, $f = \min$, $f = \max$, $f = sum$.*

Proof: Membership to NP comes from the following non-deterministic algorithm running in polynomial time: (1) guess $\omega \in 2^{Var(B)}$; (2) check that $I^f_B(\omega) \neq []$.

When $f = \cap$, NP-hardness comes from the following reduction from the satisfiability problem SAT: let Σ be any CNF formula from $PROP_{PS}$; let us associate in polynomial time the base $B = \{(true, =, \bot), (\neg\Sigma, =, \top)\}$ to Σ. If Σ is satisfiable, then it has a model ω; by construction, $I^{\cap}_B(\omega) = [\bot, \bot] \neq []$, hence B is $\cap$-consistent. If Σ is unsatisfiable, then $\neg\Sigma$ is valid; hence for every world $\omega \in 2^{PS}$, we have $I^{\cap}_B(\omega) = []$; in this case, B is not $\cap$-consistent.

When $f = +$, B is $+$-consistent if and only if B contains a 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ such that $r_1(s_1) \cap r_2(s_2) \neq []$, which can be decided in polynomial time. Similarly, when $f = \min$, $f = \max$ or $f = sum$, B is f-consistent, except when it contains a 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ such that $r_1(s_1) \cap r_2(s_2) = []$ (i.e., the interval specified by r_1, s_1, r_2, s_2 is empty). Accordingly, when interval emptiness can be decided in polynomial time, this is also the case of f-consistency for those functions f.

□

Theorem 2 *f-*POSSIBILITY *is in* P.

Proof: Trivial since $I^f_B(\omega)$ (as a normal interval) can be computed in polynomial time when f is a linear time function.

□

Theorem 3 *f-*EQUIVALENCE *is* coNP*-complete.*

Proof: Membership to NP of the complementary problem comes from the following non-deterministic algorithm running in polynomial time: (1) guess $\omega \in 2^{Var(B_1) \cup Var(B_2)}$; (2) check that $I^f_{B_1}(\omega) \neq I^f_{B_2}(\omega)$.

Hardness comes from the following reduction from the unsatisfiability problem UNSAT: let Σ be any CNF formula from $PROP_{PS}$; to Σ let us associate in polynomial time the bases $B_1 = \{(\Sigma, =, \bot)\}$ and $B_2 = \{(\Sigma, =, \top)\}$. If Σ is satisfiable, then it has a model ω; by construction, $I^f_{B_1}(\omega) = [\bot, \bot]$ and $I^f_{B_2}(\omega) = [\top, \top]$, hence B_1 and B_2 are not f-equivalent. If Σ is unsatisfiable, then for any world $\omega \in 2^{PS}$, we have $I^f_{B_1}(\omega) = I^f_{B_2}(\omega) = n_f$; in this case, B_1 and B_2 are f-equivalent.

□

Theorem 4 *(f, g)-*ORDER-EQUIVALENCE *is* coNP*-complete*

Proof: Membership to NP of the complementary problem comes from the following non-deterministic algorithm running in polynomial time: (1) guess $\omega_1, \omega_2 \in 2^{Var(B_1) \cup Var(B_2)}$; (2) check that one of the following statements hold:

- ω_1 is f-possible given B_1 and ω_1 is not g-possible given B_2,
- ω_1 is not f-possible given B_1 and ω_1 is g-possible given B_2,
- ω_1 and ω_2 are f-possible given B_1 and g-possible given B_2 and
  - · $\omega_1 \succ^f_{B_1} \omega_2$ and $\omega_1 \not\succ^g_{B_2} \omega_2$ or
  - · $\omega_1 \sim^f_{B_1} \omega_2$ and $\omega_1 \not\sim^g_{B_2} \omega_2$.

Hardness comes from the following reduction from the unsatisfiability problem UNSAT, and holds even in the restricted case $f = g$ (and $S_1 = S_2$); let Σ be any CNF formula from $PROP_{PS}$ such that $Var(\Sigma) = \{x_1, \ldots, x_n\}$; to Σ let us associate in polynomial time the bases $B_1 = \{(\Sigma \wedge new, =, \bot), (\Sigma \wedge \neg new, =, \top)\}$ and $B_2 = \{(\Sigma \wedge new, =, \top), (\Sigma \wedge \neg new, =, \bot)\}$, where new is a fresh atom from $PS \backslash Var(\Sigma)$. If Σ is satisfiable, then it has a model ω over $\{x_1, \ldots, x_n\}$, which can be extended to a model ω_1 of $\Sigma \wedge new$ (resp. a model ω_2 of $\Sigma \wedge \neg new$) by requiring that $\omega_1(new) = 1$ (resp. $\omega_2(new) = 0$); by construction, $I^f_{B_1}(\omega_1) = [\bot, \bot]$ and $I^f_{B_1}(\omega_2) = [\top, \top]$, while $I^f_{B_2}(\omega_1) = [\top, \top]$ and $I^f_{B_2}(\omega_2) = [\bot, \bot]$. Accordingly, we have $\omega_2 \succ^f_{B_1} \omega_1$ and $\omega_1 \succ^f_{B_2} \omega_2$, hence B_1 is not (f, g)-order-equivalent to B_2. In the remaining case (i.e., when Σ is unsatisfiable), we have that for any pair of worlds ω_1 and ω_2, $I^f_{B_1}(\omega_1) = I^f_{B_1}(\omega_2) = I^f_{B_2}(\omega_1) = I^f_{B_2}(\omega_2) = n_f$, hence B_1 is (f, g)-order-equivalent to B_2.

□

Theorem 5 *f-COMPARISON(op) is in* P.

Proof: Obvious given that each normal interval $I^f_B(\omega_1)$ and $I^f_B(\omega_2)$ can be computed in polynomial time when f is a linear time function.

□

Theorem 6 *f-NON-DOMINANCE is* coNP*-complete.*

Proof: As to membership, we consider the complementary problem and show that it is in NP thanks to the following non-deterministic algorithm running in polynomial time: (1) guess $\omega' \in 2^{Var(B)}$; (2) compute $I^f_B(\omega)$ and $I^f_B(\omega')$; (3) check that $I^f_B(\omega') \neq []$ and that $\omega' \succ^f_B \omega$.

Hardness comes from the following reduction from the unsatisfiability problem UNSAT: let Σ be any CNF formula from $PROP_{PS}$; to Σ let us associate in polynomial time the base $B = \{(\neg(\Sigma \wedge new), =, \bot), (\Sigma \wedge new, =, \top)\}$ (where new is a fresh atom from $PS \setminus Var(\Sigma)$), and any world ω over $2^{Var(\Sigma) \cup \{new\}}$ such that $\omega(new) = 0$; by construction, we have $I^f_B(\omega) = [\bot, \bot]$. If Σ is satisfiable, then it has a model ω' such that $\omega'(new) = 1$; by construction, we have $I^f_B(\omega') = [\top, \top]$. As a consequence, we have $\omega' \succ^f_B \omega$: ω is dominated w.r.t. $\succ^f_B$. Finally, if Σ is unsatisfiable, then B is f-equivalent to the base $\{(true, =, \bot)\}$, which interprets all the worlds in the same way: ω is not dominated w.r.t. $\succ^f_B$.

□

Theorem 7 *f-CAND-OPT-SAT is in Σ^p_2. It is both* NP*-hard and* coNP*-hard (hence it is not in* NP $\cup$ coNP *unless the polynomial hierarchy collapses). For more specific cases we have the following results:*

i. If $card(S) \geq card(B)$, f-CAND-OPT-SAT is Θ^p_2-hard.
ii. Let P^f_B be the set of potential (normal) intervals given B and f defined as the subset of Int_S given by $\{f(\{r_1(s_1) \cap r_2(s_2) \mid (\phi, r_1, s_1, r_2, s_2) \in B'\} \mid B' \subseteq B)$. If a superset P of P^f_B can be (explicitly) computed in polynomial time in the size of B, then f-CAND-OPT-SAT is in Θ^p_2.
iii. If B can be turned in polynomial time into a (f, sum)-order-equivalent base B' over PS and S' s.t. every 5-tuple $(\phi, r_1, s_1, r_2, s_2)$ satisfies $r_1 =\geq$ and $r_2 =\leq$, and S' is a closed interval of $\mathbb{N}$ containing 0, then sum-CAND-OPT-SAT is in Δ^p_2.

Proof:

- Membership to Σ^p_2 comes from the following non-deterministic polytime algorithm using an NP-oracle: (1) guess $\omega \in 2^{Var(B)}$; (2) check that ω is f-possible; (3) check that ω is not dominated w.r.t. $\succ^f_B$ using one call to an NP-oracle; (4) check that $\omega \models \phi$.
- NP-hardness comes from the following reduction from the satisfiability problem SAT: let Σ be any CNF formula from $PROP_{PS}$; to Σ let us associate in polyno-

mial time the base $B = \{(true, =, \top)\}$ and the formula $\phi = \Sigma$. By construction, Σ is satisfiable if and only if there exists a undominated world ω w.r.t. $\succ^f_B$ in the set of all worlds f-possible given B such that $\omega \models \phi$. Indeed, any model ω of Σ does the job.

- coNP-hardness comes from a straightforward polynomial reduction from f-NON-DOMINANCE: ω is not dominated w.r.t. $\succ^f_B$ if and only if there exists a undominated world ω w.r.t. $\succ^f_B$ in the set of all worlds f-possible given B such that $\omega \models \phi$ with $\phi = \omega$.
- More specific cases:

i. When $card(S) \geq card(B)$, Θ^p_2-hardness comes from a polynomial reduction from PARITY-SAT [26]: given a sequence $\phi_1, \ldots, \phi_n$ of formulae from $PROP_{PS}$ such that for all $i \in \{1, \ldots, n-1\}$, if ϕ_i is unsatisfiable then ϕ_{i+1} is unsatisfiable, is the maximum index i such that ϕ_i is satisfiable an odd number? $n = 2m$ is assumed even without loss of generality. The reduction is as follows: to $\phi_1, \ldots, \phi_n \in PROP_{PS}$, let us associate in polynomial time the base $B = \{(\phi_i \wedge new_i \wedge \bigwedge_{j=1\ldots n, j\neq i} \neg new_j, =, s_i) \mid i \in \{1, \ldots, n\}\}$ where $\{s_1, \ldots, s_n\} \subseteq S$ is such that $\forall i, j \in \{1, \ldots, n\}$, if $i \leq j$ then $s_i \leq s_j$, and the formula $\phi = \bigvee_{j=0}^{m-1} \phi_{2j+1}$. By construction, the maximum index i such that ϕ_i is satisfiable is odd if and only if there exists a model ω of ϕ which is not dominated w.r.t. $\succ^f_B$.

ii. Let us now show the membership of f-CAND-OPT-SAT to Θ^p_2 in a restricted case. Roughly, the approach consists in determining an interval i_{max} of Int_S which is maximal w.r.t. $>_{Int_S}$ and such that there exists a world ω satisfying $I^f_B(\omega) = i_{max}$; this is done using binary search and an NP-oracle; then it is enough to check using one call to an NP-oracle that there exists a model ω of ϕ s.t. $I^f_B(\omega) \sim_{Int_S} i_{max}$.

The difficulty here lies in binary searching since $>_{Int_S}$ is not necessarily complete; especially, it can be the case that i_{max} is not unique. In order to overcome it, we refrain from considering $>_{Int_S}$ directly, but a complete, strict ordering $\succ$ closely related to it. Formally, let $\succ$ be the binary relation over the set of all non-empty intervals from Int_S defined by $i_1 \succ i_2$ if and only if $l(i_1) > l(i_2)$ or ($l(i_1) = l(i_2)$ and (($lc(i_1)$ and not $lc(i_2)$)) or $u(i_1) > u(i_2)$ or ($u(i_1) = u(i_2)$ and $rc(i_1)$ and not $rc(i_2)$))). For any non-empty subset E of Int_S consisting of non-empty intervals, $max(E, \succ)$ is a singleton $\{i^E_{max}\}$. It is obvious that $\forall i_1, i_2 \in E$, if $i_1 >_{Int_S} i_2$ then $i_1 \succ i_2$. We also have $max(E, >_{Int_S}) = \{i \in E \mid i \sim_{Int_S} i^E_{max}\}$ (see Lemma 1 on the appendix for the proof).

One can now design a polynomial time algorithm SOLVE-f-CAND-OPT-SAT-1 for deciding f-CAND-OPT-SAT (under the requirements given in the proposition), using a logarithmic number of calls to an NP-oracle (please see Algorithm 1 on the appendix). It mainly consists in binary searching $\{i^E_{max}\}$ in the set of potential intervals (or the given superset of it); here, $E = \{i \in P \setminus \{[]\} \mid \exists \omega \in 2^{Var(B) \cup Var(\phi)}$ s.t. $I^f_B(\omega) = i\}$ (for the definition of rank used in the allgorithm, please see Subsection 3.1).

The evaluation of the condition of each conditional in SOLVE-f-CAND-

OPT-SAT-1 requires one call to an NP-oracle. The total number of calls to such an oracle is thus upper bounded by $\lceil log_2\ Card(P)\rceil + 1$; the result then follows immediately since $Card(P)$ is upper bounded by $p(|B|)$ where p is a polynomial such that $|P| = p(|B|)$.

iii. Let us now finally show the membership of f-CAND-OPT-SAT to Δ_2^p in a second restricted case. Roughly, the algorithm SOLVE-f-CAND-OPT-SAT-2 used to prove it consists in first computing B', then in binary searching $m = l(i^E_{max})$ with $E = \{i \in Int_{S'} \backslash \{[]\} \mid \exists \omega \in 2^{Var(B') \cup Var(\phi)}$ s.t. $I^g_{B'}(\omega) = i\}$, in the interval $[0, M = \sum_{(\phi, \geq, s_1, \leq, s_2) \in B'} s_2] \subseteq S'$ (please see Algorithm 2 on the appendix). Indeed, for any interval $i \in Int_{S'}$, we have that $i \sim_{Int_{S'}} i^E_{max}$ if and only if $m \in i$.

The evaluation of the condition of each conditional in SOLVE-f-CAND-OPT-SAT-2 requires one call to an NP-oracle. The total number of calls to such an oracle is thus upper bounded by $2 \times \lceil log_2\ M + 1\rceil + 1$. Finally, the number of bits in the binary representation of any s_2 in B' is upper bounded by $p(|B|)$ where p is a polynomial such that $|B'| = p(|B|)$; similarly, the cardinal of B' is upper bounded by $p(|B|)$, and as a consequence the number of bits in the binary representation of M is upper bounded by $2 \times p(|B|)$; it comes that the value of M is upper bounded by $2^{2 \times p(|B|)}$; thus, the total number of calls to an NP-oracle in SOLVE-f-CAND-OPT-SAT-2 is upper bounded by $2 \times \lceil log_2\ (2^{2 \times p(|B|)}) + 1\rceil + 1 = 2 \times \lceil 2 \times p(|B|) + 1\rceil + 1$, hence by a polynomial in the input size, and the result follows.

□

To conclude with the complexity results, observe that the set $\{[s_1, s_2], [s_1, s_2), (s_1, s_2], (s_1, s_2) \mid (\exists r, s(\phi_1, r_1, s_1, r, s) \in B$ or $s_1 \in \{\bot, \top\})$ and $(\exists r, s(\phi_2, r, s, r_2, s_2) \in B$ or $s_2 \in \{\bot, \top\})\}$ is a superset of both $P_B^{\cap}$, P_B^{+}, P_B^{min} and P_B^{max} and it can be computed in time polynomial in the size of B; this shows that $\cap$-CAND-OPT-SAT, $+$-CAND-OPT-SAT, min-CAND-OPT-SAT and max-CAND-OPT-SAT are in Θ_2^p.

In light of our results, it turns out that several key decision problems when dealing with preferences (f-COMPARISON(op), f-POSSIBILITY) remain tractable when interval orders are represented by weighted bases, while some other key decision problems (f-NON-DOMINANCE, f-EQUIVALENCE, (f, g)-ORDER-EQUIVALENCE, f-CONSISTENCY) become “mildly” harder (i.e., at the first level of the polynomial hierarchy) than the corresponding problems based on explicit or interval representations (which are tractable). Remember that a decision problem is at the first level of PH if and only if it belongs to Δ_2^p and it is either NP-hard or coNP-hard (for more details on the polynomial hierarchy please see Subsection 2.3). This appears as the price to be paid for the gain in spatial efficiency offered by our representation languages. Nevertheless, this complexity shift at the problem level does not imply a complexity shift at the instance level, i.e., when actual runtimes are considered; indeed, as explained before, the sizes of the explicit representations of some interval orders are exponential in the size of some weighted bases representations of the same interval orders. Hence, a (deterministic) algorithm running in (simple) exponential time for deciding any problem among f-NON-DOMINANCE, f-EQUIVALENCE, (f, g)-ORDER-EQUIVALENCE, f-CONSISTENCY can easily prove more efficient on some instances than a polytime algorithm for the "corresponding problem", i.e., when the explicit representation of an interval order is considered instead of a weighted base one.

We do not know whether f-CAND-OPT-SAT is at the first level of the polynomial hierarchy or at the second level in the general case but our results show it at the first level of the polynomial hierarchy for many interesting cases.

Finally, our results show that the additional expressive power offered by our approach does not lead to a complexity shift compared to the penalty-based approach to complete preorders representation. Indeed, the complexity results for f-POSSIBILITY, f-COMPARISON(op) and f-NON-DOMINANCE as given above coincide with the complexity results for the corresponding problems reported in [17]. In contrast to our framework, the complexity of CAND-OPT-SAT in the penalty-based approach is in Δ_2^p while this is not ensured in our approach; the difficulty comes from the fact that not only the scale is not necessarily numerical in our setting, but it is not part of the input of the decision problem (hence there is no way to compute a notion of "mean" value, which is required for binary searching).

6 Conclusion

In this paper, we have shown how interval orders can be encoded as weighted bases, a subject, to our knowledge, that has not been studied before. Our presentation is simple and general in the sense that aggregation functions that can be used are not fixed but just have to satisfy some basic properties. We gave some simple examples of the use of such weighted bases when different aggregation functions are considered. Among other things, we have shown that all interval orders can be represented using some weighted bases and we have identified the complexity of a number of decision problems pertaining to the exploitation of compactly represented interval orders.

This work calls for a number of directions of future work. One of them consists in designing compact representations for other preference relations, including partially ordered intervals and PQI interval orders. PQI interval orders are preference structures with three relations P (strict preference relation), Q (weak preference relation) and I (indifference). They have been introduced and characterized by Tsoukiàs and Vincke [24,25]. PQI interval orders have an interval representation: strict preference holds when one interval is completely to the right of the other one, weak preference holds when two intervals have a non-empty intersection without inclusion and finally indifference holds in case of inclusion. Concerning their numerical representation Ngo The and Tsoukiàs ([20]) have given two algorithms, the first one in $\mathcal{O}(n^2)$ determines a general representation of these structures and the second one in $\mathcal{O}(n)$ minimizes the representation given by the first algorithm. Our weighted base representation appears appropriate for these structures. It is sufficient to determine the three relations P, Q and I as we did for the preference and indifference relations in Definition 10.

A second perspective consists in investigating further the expressiveness and spatial efficiency issues within the family of representation languages we pointed out. Indeed, the choice of S and f has a clear impact on both issues in the general case. It turns out that some spatial efficiency results from [5] for complete preorders can be easily extended to the case of interval orders (for instance, when a numerical scale is considered, the inability of compactly encoding some exponentially long P_R-chains using min or max can be exploited to show the corresponding languages less succinct than those for which sum is used). It would prove valuable to determine the expressiveness landscape and the spatial efficiency landscape for various choices of the parameters S and f.

Acknowledgements

The authors are very grateful to the anonymous reviewers for many helpful comments and suggestions. They have been partly supported by the ANR project PHAC (ANR-05-BLAN-0384).

Appendix

Lemma 1 *Let E be any non-empty subset of Int_S consisting of non-empty intervals. We have:*

$$max(E, >_{Int_S}) = \{i \in E \mid i \sim_{Int_S} i^E_{max}\}.$$

Proof:

i) $\forall i \in Int_S$, if $i \nsim_{Int_S} i^E_{max}$ then $i \notin max(E, >_{Int_S})$: it is easy to see that if $i <_{Int_S} i^E_{max}$ then $i \notin max(E, >_{Int_S})$. On the other hand $i >_{Int_S} i^E_{max}$ is not possible since by the definition of i^E_{max} we have $l(i^E_{max}) > l(i)$ or ($l(i^E_{max}) = l(i)$ and ((($lc(i^E_{max})$ and not $lc(i)$) or $u(i^E_{max}) > u(i)$ or ($u(i^E_{max}) = u(i)$ and $rc(i^E_{max})$ and not $rc(i)$)))). And $u(i^E_{max})$ being greater than or equal to $l(i^E_{max})$, it is not possible to have $l(i) > u(i^E_{max})$ when $rc(i^E_{max})$ or $l(i) \geq u(i^E_{max})$ when not $rc(i^E_{max})$.

ii) $\forall i \in Int_S$, if $i \sim_{Int_S} i^E_{max}$ then $\forall j$ we have $j \sim_{Int_S} i$ or $j <_{Int_S} i$: we show that it is impossible to have $j >_{Int_S} i$. Suppose that there exists j such that $j >_{Int_S} i$. Then we have $l(j) > u(i)$ if $rc(i)$ (or $l(j) \geq u(i)$ if $\neg rc(i)$). On the other hand since $i \sim_{Int_S} i^E_{max}$ we have $l(i) < l(i^E_{max}) < u(i)$. So we will have $l(i) < l(i^E_{max}) < u(i) < l(j)$, which is in contradiction with the definition of i^E_{max}.

□

Algorithm 1 : Polytime algorithm for f-CAND-OPT-SAT using logarithmically many calls to an NP-oracle

```
procedure SOLVE-$f$-CAND-OPT-SAT-1
Data : A weighted base $B$ over $PS$ and $S$ and a formula $\phi \in PROP_{PS}$
Result : 1 if a model of $\phi$ undominated w.r.t. $\succ^f_B$ exists, 0 otherwise
begin
    Compute $P = P^f_B$ (or a superset of it) in time polynomial in $|B|$ ;
    $O$ = the set obtained by removing from $P$ the empty interval [] (if present) ;
    Sort $O$ w.r.t. $\succ$ ;
    while $O$ is not a singleton $\{i_{max}\}$ do
        $i$ = the interval of rank $\lfloor Card(O)/2 \rfloor$ in $O$ ;
        if $\exists \omega \in 2^{Var(B) \cup Var(\phi)}$ such that $I^f_B(\omega) \in O$ and $I^f_B(\omega) \succ i$ then
            Remove from $O$ every interval $j$ such that $I^f_B(\omega) \succ j$
        else
            Remove from $O$ every interval $j$ such that $j \succ i$
    if $\exists \omega \in 2^{Var(B) \cup Var(\phi)}$ such that $\omega \models \phi$ and $I^f_B(\omega) \sim_{Int_S} i_{max}$ then
        return 1
    else
        return 0
end
```

Algorithm 2 : Polytime algorithm for f-CAND-OPT-SAT using an NP-oracle

```
procedure SOLVE-$f$-CAND-OPT-SAT-2
Data : A weighted base $B$ over $PS$ and $S$ and a formula $\phi \in PROP_{PS}$
Result : 1 if a model of $\phi$ undominated w.r.t. $\succ_B^f$ exists, 0 otherwise
begin
    Compute $B'$ over $PS$ and a closed interval $S'$ of $\mathbb{N}$ such that $B'$ is $(f, sum)$-order-equivalent to $B$ ;
    $l = 0$ ;
    $u = \sum_{(\phi,\geq,s_1,\leq,s_2)\in B'} s_2$ ;
    while $l \neq u$ do
        if $\exists\omega \in 2^{Var(B')\cup Var(\phi)}$ such that $I_{B'}^{sum}(\omega) \neq []$ and $l(I_{B'}^{sum}(\omega)) > m$ then
            $l = m + 1$
        else
            if $\exists\omega \in 2^{Var(B')\cup Var(\phi)}$ such that $m \in I_{B'}^{sum}(\omega)$ then
                $l = u = m$
            else
                $u = m$
    if $\exists\omega \in 2^{Var(B')\cup Var(\phi)}$ such that $\omega \models \phi$ and $m \in I_{B'}^{sum}(\omega)$ then
        return 1
    else
        return 0
end
```

References

[1] F. Bacchus and A.J. Grove. Utility independence in a qualitative decision theory. In *Proc. of KR'96*, pages 542–552, 1996.

[2] S. Bistarelli, H. Fargier, U. Montanari, F. Rossi, Th. Schiex, and G. Verfaillie. Semiring-based CPSs and valued CSPs: Framework, properties and comparison. *Constraints*, 4(3):199–240, 1999.

[3] C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. CP-nets: a tool for representing and reasoning with conditional ceteris paribus statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004.

[4] Y. Chevaleyre, U. Endriss, and J. Lang. Expressive power of weighted propositional formulas for cardinal preference modeling. In *Proc. of KR'06*, pages 145–152, 2006.

[5] S. Coste-Marquis, J. Lang, P. Liberatore, and P. Marquis. Expressive power and succinctness of propositional languages for preference representation. In *Proc. of KR'04*, pages 203–212, 2004.

[6] A. Darwiche and P. Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.

[7] J.-P. Doignon. Thresholds representation of multiple semiorders. *SIAM Journal on Algebraic and Discrete Methods*, 8:77–84, 1987.

[8] J.-P. Doignon. Sur les représentations minimales des semiordres et des ordres d'intervalles. *Mathématiques et Sciences Humaines*, 101:49–59, 1988.

[9] J. Doyle and M.P. Wellman. Preferential semantics for goals. In *Proc. of AAAI'91*, pages 698–703, 1991.

[10] P.C. Fishburn. *Interval orders and interval graphs*. John Wiley and Sons, 1985.

[11] P.C Fishburn. Generalisations of semiorders: a review note. *Journal of Mathematical Psychology*, 41:357–366, 1997.

[12] Ch. Gonzales and P. Perny. GAI networks for utility elicitation. In *Proc. of KR'04*, pages 224–234, 2004.

[13] G. Isaak. Bounded discrete representations of interval orders. *Discrete Applied Mathematics*, 44:157–183, 1993.

[14] D.H Krantz, R.D Luce, P. Suppes, and A. Tversky. *Foundations of measurement*, volume 1: Additive and polynomial representations. Academic Press, New York, 1971.

[15] D.H Krantz, R.D Luce, P. Suppes, and A. Tversky. *Foundations of measurement*, volume 2: Geometrical, threshold and probabilistic representations. Academic Press, New York, 1989.

[16] C. Lafage and J. Lang. Logical representation of preferences for group decision making. In *Proc. of KR'00*, pages 457–468, 2000.

[17] J. Lang. Logical preference representation and combinatorial vote. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):37–71, 2004.

[18] R.D. Luce. Semi-orders and a theory of utility discrimination. *Econometrica*, 24, 1956.

[19] R.D Luce, D.H Krantz, P. Suppes, and A. Tversky. *Foundations of measurement*, volume 3: Representation, axiomatisation and invariance. Academic Press, New York, 1990.

[20] A. Ngo The and A. Tsoukiàs. Numerical representation of PQI interval orders. *Discrete Applied Mathematics*, 147:125–146, 2005.

[21] Ch. Papadimitriou. *Computational complexity*. Addison–Wesley, 1994.

[22] M. Pirlot and Ph. Vincke. *Semi Orders*. Kluwer Academic, Dordrecht, 1997.

[23] D. Scott and P. Suppes. Foundational aspects of theories of measurement. *Journal of Symbolic Logic*, 23:113–128, 1958.

[24] A. Tsoukiàs and Ph. Vincke. Double threshold orders: A new axiomatization. *Journal of Multi-criteria Decision Analysis*, 7:285–301, 1998.

[25] A. Tsoukiàs and Ph. Vincke. A characterization of PQI interval orders. *Discrete Applied Mathematics*, 127(2):387–397, 2003.

[26] K.W. Wagner. Bounded query classes. *SIAM Journal on Computing*, 19(5):833–846, 1990.

[27] N. Wiener. A contribution to the theory of relative position. *Proc. of Cambridge Philosophical Society*, 17:441–449, 1914.