# Multi-Criteria Decision Making with Existential Rules using Repair Techniques

Madalina Croitoru[1], Pierre Bisquert[2], Christos Kaklamanis[3,4], Nikos Karanikolas[4], Rallou Thomopoulos[2], and Bruno Yun[1]

[1] University of Montpellier, France
{croitoru,yun}@lirmm.fr
[2] INRA Iate/INRIA GraphIK, Montpellier, France
{pierre.bisquert,rallou.thomopoulos}@inra.fr
[3] Computer Technology Institute and Press "Diophantus" (CTI), Patras, Greece
[4] Department of Computer Engineering and Informatics, University of Patras, Greece
{kakl,nkaranik}@ceid.upatras.gr

**Abstract.** The central problem in multi-criteria decision making is to reach an acceptable decision aggregating preferences over multiple criteria. In this paper, we explain how to benefit from the reasoning capabilities of existential rules for modelling a multi-criteria decision making problem as an inconsistent knowledge base. The repairs of this knowledge base represent the maximally consistent point of views and inference strategies can be used for decision making.

## 1 Introduction

The way to reach a group's decision is a very complex task and depends on the nature of the decision problem. A common way for agents to take their decisions is to analyze the problem in different criteria which correspond to the different aspects that they consider to be important for the decision. This can be depicted as the multi-criteria decision making (MCDM) problem, where the set of agents corresponds to the decision makers that have preferences over multiple criteria on a set of alternatives. MCDM has been an active research area since the 60s when B. Roy ([11]) introduced the class of ELECTRE methods for aggregating preferences expressed on multiple criteria. Its development continued with the foundational "outranking methods" ([22]) and has been since an active research field. Recently these techniques have been used for recommender systems, whose objective is to recommend a solution given the decision makers' evaluation on different criteria of the alternative options. Many aggregation strategies for solving the group recommender systems can be applied to the multi-criteria ones [20].

In this paper we take a different research avenue and see the MCDM problem from a knowledge representation and reasoning point of view. The advantage of doing so is two fold. First, it allows for the use of expressive languages for describing the decision problem (and their subsequent reuse in applications). Second, it paves the way for synergies between the two fields. MCDM could thus benefit from recent advances on explanation and user interaction developed within the techniques we employ in this paper.

Concretely, we propose to see the problem description of a MCDM problem as an inconsistent knowledge base expressed using existential rules. The inconsistency will be addressed using repair techniques [7], a state of the art method of reasoning in presence of inconsistency that outputs the consistent subsets of the knowledge base maximal with respect to set inclusion. The reasoning is then performed on these subsets (also called repairs of the knowledge base).

Classically, in the MCDM problem, we have a set of alternative options (denoted by the set of constants $\mathcal{C}_A$) and a set of agents/decision makers $\mathcal{I}$. Each alternative expresses a set of alternatives (denoted by $\mathcal{C}_{A,\mathcal{I}}$) and a set of criteria (denoted by the set of predicates $\mathcal{P}_{C,\mathcal{I}}$). The set of all criteria predicates, i.e., the set of criteria given by all the decision makers is denoted by $\mathcal{P}_C$. The decision makers are called to express their preferences concerning each of these alternatives by taking into account the criteria.

We will use the knowledge bases produced by each decision maker in order to compute the set of repairs $\boldsymbol{E}$ which aggregates the different decision makers preferences. The inconsistency tolerant semantics will allow to reason on repairs. In this paper we do not focus on the inconsistency tolerant reasoning part but rather on the modelling as an inconsistent knowledge base. This is due to the fact that our contribution lays in the modelling aspects rather than reasoning aspects (that employ state of the art algorithms). Let us first present the logical language used throughout the paper.

## 2 Background notions

In this section we define the logical language employed in this paper. Please note that the notions below represent classical logical notions with some minor changes to reflect the decision making setting, in which we place ourselves in. We consider *the positive existential* fragment of first-order logic $\texttt{FOL}(\exists, \wedge)$ [14, 8]. Its language $\mathcal{L}$ is composed of formulas built with the usual quantifiers $(\exists, \forall)$ and *only* two connectors: implication $(\rightarrow)$ and conjunction $(\wedge)$.

We consider usual first-order vocabularies with constants but no other function symbols. A vocabulary is a pair of two disjoint sets $\mathcal{V} = (\mathcal{P}, \mathcal{C})$, where $\mathcal{P}$ is a finite set of predicates and $\mathcal{C}$ is a set of constants.

The following refining on the vocabulary definition is proper to the way we formalize the decision problem involving alternatives and values on the decision criteria over these alternatives. More precisely:

- The set of predicates $\mathcal{P}$ is partitioned in two disjoint sets $\mathcal{P} = \mathcal{P}_R \cup \mathcal{P}_C$ with $\mathcal{P}_R \cap \mathcal{P}_C = \emptyset$. $\mathcal{P}_C$ represents the criteria predicates we consider for decision making while $\mathcal{P}_R$ represents the other predicates (relations) which are used for describing the world in general (and thus not considered directly by the decision process).
- The set of constants $\mathcal{C}$ is partitioned into four pairwise disjoint sets $\mathcal{C} = \mathcal{C}_A \cup \mathcal{C}_\mathcal{I} \cup \mathcal{C}_V \cup \mathcal{C}_C$ and for all $x, y \in \{A, \mathcal{I}, V, C\}$ s.t. $x \neq y$, $\mathcal{C}_x \cap \mathcal{C}_y = \emptyset$. The set $\mathcal{C}_A$ is representing the constants that are naming the various alternatives involved in the decision process. The set $\mathcal{C}_\mathcal{I}$ represents the constants that are naming the agents/decision makers of $\mathcal{I}$ that are taking part in the decision-making problem. Let $\mathcal{C}_V = \bigcup_{c \in \mathcal{P}_C} \mathcal{C}_V^c$, where a set $\mathcal{C}_V^c$ represents the different values the criteria

$c \in \mathcal{P}_C$ can have. Therefore the interpretation of a predicate $c$ in $\mathcal{P}_C$ will only take values from $\mathcal{C}_A$ and their corresponding $\mathcal{C}_V^c$. Note that, the interpretation of an $n$-ary predicate symbol is a set of $n$-tuples of elements of the domain of discourse.The set $\mathcal{C}_C$ represents the other constants used eventually by $\mathcal{P}_R$. The domain of interpretation of $\mathcal{P}_R$ is $C_A \cup C_C$.

- Furthermore, for each predicate $c$ in $\mathcal{P}_C$ we define on the values of its interpretation $C_A \times \mathcal{C}_V^{c\,(n-1)}$, where $n$ is the arity of the predicate, a total order $\gg_{C_A \times \mathcal{C}_V^c}$. By abuse of notation, when the set of alternatives is given, we denote the preference $\gg_{\mathcal{C}_V^c}$.

A term $t$ over $\mathcal{V}$ is a constant or a variable; different constants represent different values (unique name assumption). In practice, when each decision maker gives her vocabulary she gives the set of alternatives, the set of criteria and the set of relations. Then, as we will see later, each decision maker gives for each alternative its profile, with the set of profiles for all alternatives being the Profile of the decision maker corresponding to her set of alternatives and criteria.

*Example 1.* For instance, for a decision maker $I \in C_{\mathcal{I}}$, we consider three alternatives $\mathcal{C}_{A,\mathcal{I}} = \{A_1, A_2, A_3\}$ and the criteria $\mathcal{P}_{C,\mathcal{I}} = \{c_1, c_2\}$ where $c_1$ is a binary predicate and $c_2$ is a ternary predicate. Please note that both predicates represent an alternative who has some values - either by means of a binary predicate or by the means of a ternary (or more). We do not represent here, by the means of $\mathcal{P}_{C,\mathcal{I}}$ that a criterion is preferred to the other or two values are preferred. This is done outside the FOL formalism.

$\mathcal{C}_V = \{V_1, V_2, V_3, V_4, V_5\}$ and more precisely:

- $\mathcal{C}_V^{c_1} = \{V_1, V_2\}$ and $\mathcal{C}_V^{c_2} = \{V_3, V_4, V_5\}$. This signifies that the $c_1$ predicate can only take $\{V_1, V_2\}$ as values and the $c_2$ predicate can only take $\{V_3, V_4, V_5\}$ as values.
- $V_1 \gg_{\mathcal{C}_V^{c_1}} V_2$ and $V_3 \gg_{\mathcal{C}_V^{c_2}} V_4 \gg_{\mathcal{C}_V^{c_2}} V_5$

We also consider $\mathcal{P}_R = \{p_1, p_2, p_3\}$ with $p_1$ and $p_2$ unary predicate and $p_3$ binary and $C_C = \{C_1, C_2, C_3\}$.

An **atomic formula** (or atom) over $\mathcal{V}$ is of the form $p(t_1, ..., t_n)$ where $p \in \mathcal{P}$ is an n-ary predicate, and $t_1, ..., t_n$ are terms. A ground atom is an atom with no variables. A conjunction of atoms is called a *conjunct*. A conjunction of ground atoms is called a *ground conjunct*. A variable in an atom is free if it is not in the scope of any quantifier. A formula is *closed* if it has no free variables. A closed formula is called a *sentence*. Factual knowledge (a fact) about the world is represented by *ground atoms*. In the Existential Rules framework this concept has been extended so that a fact on $\mathcal{V}$ is the existential closure of a conjunction of atoms over $\mathcal{V}$ [8]. Let $F$ be a fact, we denote by $terms(F)$ (resp. $vars(F)$) the set of terms (resp. variables) that occur in $F$.

In the following, we consider a special kind of fact called **profile**. This will allow us to give, for each alternative, the set of criteria we consider for the decision process along to their corresponding values. A *profile* for alternative $A \in \mathcal{C}_A$ and for decision maker $I \in \mathcal{C}_{\mathcal{I}}$, denoted $profile_I(A)$, is a conjunction of facts of the form $p_0(A) \wedge \bigwedge_{1 \leq i \leq n} p_i(A, \boldsymbol{t})$ with $A$ representing an alternative (i.e., $A \in \mathcal{C}_A$), $p_0$ an unary predicate in $\mathcal{P}_R$ representing the type of the alternative, $\{p_i | 1 \leq i \leq n\}$ a set of criteria that apply for the alternative $A$ and $\boldsymbol{t}$ a vector of the other terms except $A$ of the respective criterion predicate. Note that, in a decision problem multiple types of alternatives can be proposed as the solution and hence, the need to define $p_0$.

*Example 1 (cont.).*

For the three alternatives $\mathcal{C}_{A,\mathcal{I}} = \{A_1, A_2, A_3\}$ and the predicates and constants defined in the previous examples we have three profiles which are the following and correspond to each of the three alternatives:

- $profile_I(A_1) = p_1(A_1) \wedge c_1(A_1, V_1) \wedge c_2(A_1, V_3, V_4)$
- $profile_I(A_2) = p_1(A_2) \wedge c_1(A_2, V_1) \wedge c_2(A_2, V_4, V_5)$
- $profile_I(A_3) = p_1(A_3) \wedge c_1(A_3, V_2) \wedge c_2(A_3, V_3, V_5)$

Each decision maker $I \in \mathcal{C}_\mathcal{I}$ will express her set of profiles over the set of alternatives $\mathcal{C}_{A,\mathcal{I}}$. This is denoted by $Profile_I(\mathcal{C}_{A,\mathcal{I}})$. Hence, $Profile_I(\mathcal{C}_{A,\mathcal{I}}) = \bigwedge_{A_k \in \mathcal{C}_{A,\mathcal{I}}} profile_I(A_k)$. When $I$ is implicit we denote it simply by $Profile(\mathcal{C}_A) = \bigwedge_{A_k \in \mathcal{C}_{A,\mathcal{I}}} profile_I(A_k)$.

*Example 1 (cont.).*

If we consider that the profiles in the previous example were given by one decision maker then, for $\mathcal{C}_A = \{A_1, A_2, A_3\}$ then $Profile(\mathcal{C}_A) = p_1(A_1) \wedge c_1(A_1, V_1) \wedge c_2(A_1, V_3, V_4) \wedge p_1(A_2) \wedge c_1(A_2, V_1) \wedge c_2(A_2, V_4, V_5) \wedge p_1(A_3) \wedge c_1(A_3, V_2) \wedge c_2(A_3, V_3, V_5)$.

We recall the notions of *substitution* and *homomorphism* between facts to be used in order to reason about profiles or facts in general. Given a set of variables $\mathcal{X}$ and a set of terms $\mathcal{T}$, a **substitution** $\sigma$ of $\mathcal{X}$ by $\mathcal{T}$ (notation $\sigma : \mathcal{X} \rightarrow \mathcal{T}$) is a function from $\mathcal{X}$ to $\mathcal{T}$. Given a fact $F$, $\sigma(F)$ denotes the fact obtained from $F$ by replacing each occurrence of $x \in X \cap vars(F)$ by $\sigma(x)$. A **homomorphism** from a fact $F$ to a fact $F'$ is a substitution $\sigma$ of $vars(F)$ by (a subset of) $terms(F')$ such that $\sigma(F) \subseteq F'$.

A conjunctive query (CQ) has the following form: $\mathcal{Q} = \texttt{ans}(x_1, \ldots, x_k) \leftarrow B$, where $B$ (the "body" of $\mathcal{Q}$) is an existential closed atom or a conjunction of existential closed atoms, and $x_1, \ldots, x_k$ are variables that occur in $B$ and $\texttt{ans}$ is a special k-ary predicate, whose elements are used to build an answer. Given a set of facts $\mathcal{F}$, an answer to $\mathcal{Q}$ in $\mathcal{F}$ is a tuple of constants $(D_1, \ldots, D_k)$ such that there is a homomorphism $\sigma$ from $B$ to $\mathcal{F}$, with $\sigma(ans(x_1, x_2, \ldots x_k)) = \sigma(ans(D_1, D_2, \ldots D_k))$. If $k = 0$, i.e., $\mathcal{Q} = \texttt{ans}() \leftarrow B$, $\mathcal{Q}$ is called a Boolean conjunctive query, the unique answer to $\mathcal{Q}$ is the empty tuple if there is a homomorphism from $B$ to $\mathcal{F}$, otherwise there is no answer to $\mathcal{Q}$. Note that a query $\mathcal{Q}$ can be shortly referred to by its body $B$. For its simplicity, this notation will be used hereafter.

*Example 1 (cont.).*

For example, if we want to retrieve the alternatives in the profiles that have the value $V_1$ for the criterion $c_1$ then we can write the following query: $\mathcal{Q} = \texttt{ans}(x_1) \leftarrow p_1(x_1) \wedge c_1(x_1, V_1)$. The set of possible answers to the query in $Profile(\mathcal{C}_A)$ will be $\{(A_1), (A_2)\}$.

In order to represent enriched knowledge about the facts, we use rules that encode domain-specific knowledge. Rules are regarded as an ontological layer that reinforces the expressiveness of the knowledge base. Rules are logical formulae that allow us to infer new facts (conclusion) from existing facts (hypothesis). **Existential rules** [8, 13] introduce new variables in the conclusion having ability to represent unknown individuals (also known in database community as *value invention* [1]). This form of rules is also known as tuple-generating dependencies in database community [16]. We denote by $\boldsymbol{x}$ in a bold font a vector of variables. An *existential rule* (or simply a rule) is a closed formula of the form $R = \forall \boldsymbol{x} \forall \boldsymbol{y}(B \rightarrow \exists \boldsymbol{z} H)$, where $B$ and $H$ are conjuncts, with $vars(B) = \boldsymbol{x} \cup \boldsymbol{y}$, and $vars(H) = \boldsymbol{x} \cup \boldsymbol{z}$. The variables in vector $\boldsymbol{z}$ are called the existential variables of the rule $R$. $B$ and $H$ are respectively called the *body* and the *head* of $R$. We denote them respectively $body(R)$ for $B$ and $head(R)$ for $H$.

*Example 1 (cont.).* Examples of rules in the example considered before could include:

- $R_1 = \forall x(c_1(x, V_1) \rightarrow p_1(x))$
- $R_2 = \forall x(c_1(x, V_2) \rightarrow p_2(x))$
- $R_3 = \forall x(c_2(x, V_3, V_4) \rightarrow p_2(x))$
- $R_4 = \forall x \forall y(c_2(x, y, V_4) \rightarrow \exists z p_3(x, z))$

Real examples of the above rules could be that an alternative that has value $V_1$ for price is a cheap alternative ($R_1$) and an alternative that has value $V_2$ for price is an expensive alternative ($R_2$). $R_3$ might state that an alternative that has good reviews in the $V_3$ journal is an expensive alternative. The last rule might say that an alternative that has good reviews (i.e., value $V_4$) in all journals (i.e. variable $y$) will be discounted in at least one shop (i.e., the existential variable $z$).

Existential rules are more expressive than Description Logics as they can represent complex relations between individuals and overcome the "cycle on variables" [14]. Another important aspect of the existential rules framework is the possibility of having *unbounded predicate arity*, i.e, predicates with an arbitrary number of parameters. Entailment is undecidable for general existential rules [10]. However, many classes of existential rules that ensure decidability (while keeping expressiveness) have been studied (see [8]). In this paper and for practical reasons we work on such classes.

We also account for a special kind of rules called *negative constraints*, i.e., knowledge that imposes constraints about the world, also known as denial constraints in databases [13]. A negative constraint (or simply a constraint) is a rule of the form $N = \forall \boldsymbol{x}(B \rightarrow \bot)$, where $vars(B) = \boldsymbol{x}$. Negative constraints in the existential rules framework fully capture *concept disjointness* of DLs. From now on we omit quantifiers in front of formulae as there is no ambiguity.

*Example 1 (cont.).* An example of negative constraint on the vocabulary of the example considered before could be $\forall x(p_1(x) \wedge p_2(x) \rightarrow \bot)$ (for example an alternative cannot be expensive and cheap at the same time).

An individual decision maker's $I$ **knowledge base** is a tuple $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})_I$ (denoted as $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ when there is no ambiguity) of finite sets of facts (thus including profiles), rules and negative constraints respectively. Reasoning with a knowledge base of facts, rules and negative constraints is done via a mechanism called saturation of facts by the rules. In order to define saturation we need to define rule applicability on facts.

A rule $R = \forall \boldsymbol{x} \forall \boldsymbol{y}(B \rightarrow \exists \boldsymbol{z} H)$ is **applicable** [8] to a fact $F$ if there exists a homomorphism $\sigma$ from $B$ to $F$. The *application of $R$ to $F$ w.r.t. $\sigma$* produces a fact $\alpha(F, R, \sigma) = F \cup \sigma(safe(H))$, where $\mathtt{safe}(H)$ is obtained from $H$ by replacing existential variables with fresh variables (not used variables). $\alpha(F, R, \sigma)$ is said to be an **immediate derivation** from $F$. Let $F$ be a fact and $\mathcal{R}$ be a set of rules. A fact $F'$ is called an **$\mathcal{R}$-derivation** of $F$ if there is a finite sequence (called the **derivation sequence**) $\langle F_0 = F, ..., F_n = F' \rangle$ such that for all $0 \leq i < n$ there is a rule $R \subseteq \mathcal{R}$ which is applicable to $F_i$ and $F_{i+1}$ is an immediate derivation from $F_i$. The saturation operator ($C\ell$) can be seen as a *fixed-point operator* where we denote by $C\ell^*_{\mathcal{R}}(\mathcal{F})$ the saturation of $\mathcal{F}$ with respect to $\mathcal{R}$. Note that $C\ell^*_{\mathcal{R}}(\mathcal{F})$ is a finite set [9] for the classes of existential rules considered in this paper.

*Example 1 (cont.).*

Let us consider again the profiles for $\mathcal{C}_A = \{A_1, A_2, A_3\}$: $Profile(\mathcal{C}_A) = p_1(A_1) \wedge c_1(A_1, V_1) \wedge c_2(A_1, V_3, V_4) \wedge p_1(A_2) \wedge c_1(A_2, V_1) \wedge c_2(A_2, V_4, V_5) \wedge p_1(A_3) \wedge c_1(A_3, V_2) \wedge c_2(A_3, V_3, V_5)$.

$R_1$ and $R_2$ are applicable on $Profile(\mathcal{C}_A)$ yielding the new facts $p_1(A_1)$, $p_1(A_2)$ and $p_2(A_3)$. $R_3$ is also applicable on $Profile(\mathcal{C}_A)$ yielding the new fact $p_2(A_1)$. The facts $p_1(A_1)$ and $p_2(A_1)$ will trigger the negative constraint $\forall x(p_1(x) \wedge p_2(x) \rightarrow \bot)$ (thus, as we will see in the next subsection) rendering $Profile(\mathcal{C}_A)$ inconsistent.

In the next subsection we will see how to reason with facts in the presence of inconsistency using repair based methods.

## 3 Multi-criteria Decision Making as Repair Techniques

In what follows we recall the formal definition of *inconsistency* in the existential rules framework; then we introduce the subset-repairing techniques which is inspired by the work from the database community [15] and Description Logics [19, 12]. Please note that the aim of this paper is to show how the decision problem can be modelled as an inconsistent knowledge base. For this reason we will simply favor the modelling part explaining the various choices made and will not go deeper into the inconsistency tolerant semantics aspect.

**Definition 1 (Inconsistency).** *A set of facts $\mathcal{F}$ is inconsistent with respect to a set of rules $\mathcal{R}$ and negative constraints $\mathcal{N}$ (or inconsistent for short) if and only if there exists a constraint $N \in \mathcal{N}$ such that $C\ell^*_{\mathcal{R}}(\mathcal{F}) \models body(N)$.*

This means that the set of facts *violates* the negative constraint $N$ or triggers it. Correspondingly, a knowledge base $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ is inconsistent (with respect to $\mathcal{R}$ and $\mathcal{N}$) if and only if there exists a set of facts $\mathcal{F}' \subseteq \mathcal{F}$ such that $\mathcal{F}'$ is inconsistent. An alternative writing is $C\ell^*_{\mathcal{R}}(\mathcal{F}) \models \bot$.

One way to cope with inconsistency is to construct *maximal consistent subsets* of the knowledge base [23]. This corresponds to “Data Repairs” [3]. A data repair of a knowledge base $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ is a set of facts $\mathcal{F}'$ such that $\mathcal{F}'$ is consistent and there exists no consistent subset of $\mathcal{F}$ that strictly contains $\mathcal{F}'$ [19].

Since repairs are computed exclusively on the set of facts and given that the factual part of the knowledge base is the only source of inconsistency we, from now on, abuse slightly the notation and refer to $\mathcal{K}'$ by its set of facts $\mathcal{F}'$. The set of all repairs of $\mathcal{K}$ is denoted by $\mathcal{R}epair(\mathcal{K})$.

For one individual decision maker $I \in \mathcal{C}_{\mathcal{I}}$ and the set of alternatives $\mathcal{C}_A$ we use repair techniques in order to obtain one single, maximal consistent subset of $Profile(\mathcal{C}_A)$.

For a given set of alternatives $\mathcal{C}_A$ and for each of the individual decision makers $I \in \mathcal{C}_{\mathcal{I}}$, the $Profile_I(\mathcal{C}_A)$ is consistent wrt to the commonly agreed set of rules and negative constraints [5]. Note that we need to have a common structure for all the profiles and we do not consider that the preferences of an individual decision maker are inconsistent.

The first step is to construct $Profile_I(\mathcal{C}_A)$ for every decision maker $I \in \mathcal{C}_{\mathcal{I}}$.

*Example 2.* In the following example, we present a multi-criteria decision problem that will guide us throughout the modeling section. The multi-criteria decision problem illustrated is a complicated problem every young parent faces: what pushchair to buy (preferably before the baby arrives)? We have the set of alternatives $\mathcal{C}_A = \{Yoyo, Chicco, Inglesina\}$, the set of decision makers $\mathcal{C}_{\mathcal{I}} = \{M, P, J\}$, where $I \in \mathcal{C}_{\mathcal{I}}$ are constants, and the set of criteria $\mathcal{P}_c = \{price, weight, transportable\}$. Note that the criterion $transportable$ is a ternary predicate$(a, size, foldable)$. The preferences of the decision maker $M$ for all the alternatives are included in the $Profile_M(\mathcal{C}_A) = profile_M(Yoyo) \wedge profile_M(Chicco) \wedge profile_M(Inglesina)$ , where:

- $profile_M(Yoyo) = pushchair(Yoyo) \wedge price(Yoyo, 500) \wedge weight(Yoyo, 3)$ $\wedge\, transportable(Yoyo, large, yes)$
- $profile_M(Chicco) = pushchair(Chicco) \wedge price(Chicco, 300) \wedge$ $weight(Chicco, 4) \wedge transportable(Chicco, large, no)$
- $profile_M(Inglesina) = pushchair(Inglesina) \wedge price(Inglesina, 400)$ $\wedge\, weight(Inglesina, 2) \wedge transportable(Inglesina, medium, yes)$

At this step it is very important to mention the fact that each decision maker also has a (strict) total preference ordering on the set of criteria they consider in their profiles.

Inconsistencies will also arise at this step are therefore due to several factors:

- The set of criteria is not the same for all decision makers.
- The set of alternatives is not the same for all decision makers.
- The preferences of criteria are not the same for all decision makers.
- The value of an alternative for a given criterion is not the same for all decision makers.
- The subjective preferences (i.e., the evaluation ratings) given by the decision makers for an alternative on a given criterion are not the same.

[5] Please remember that we are in the OBDA case where the inconsistency can only come from facts thus we agree upon a common ontology (i.e., set of rules and negative constraints) a priori.

At this step we consider a fresh set of constants that correspond to those criteria that the decision maker $I$ considers in her aggregation function. We also need to consider the set of constants naming the decision makers, i.e., for every $c \in \mathcal{P}(c)$ the new constant name is $C^*$. We introduce two meta predicates: *consider* (a ternary predicate taking a decision maker's identifier, an alternative and a constant corresponding to the criterium name) and *preferred* (a ternary predicate stating that $I$ prefers the criterion $i$ to criterion $j$).

### 3.1 Inconsistency regarding the consideration of different criteria

Each decision maker $I \in \mathcal{C}_\mathcal{I}$ gives a set of rules of the form $\mathcal{R}_I = \{\forall A \in \mathcal{C}_A\ type(A) \wedge \mathcal{J}_{I_i} \rightarrow consider(I, A, C^*)\}$, where $C^*$ is the constant that corresponds to a criterion that $I$ considers for alternative $A$. We have that $type$ is a predicate and $A$ is in its interpretation. By $\mathcal{J}_i$, we denote the justification made by the decision maker for her rule $R_i$. Negative constraints occur when different decision makers do not consider the same criteria. Therefore negative constraints are of the form $\forall I, J \in \mathcal{C}_\mathcal{I} \forall A \in \mathcal{C}_A \forall C^* \in \mathcal{C}_C, consider(I, A, C^*) \wedge not\_consider(J, A, C^*) \wedge type(A) \wedge diff(I, J) \rightarrow \perp$. The predicate $not\_consider(I, A, C^*)$ corresponds to the case where criterion $C^*$ is not considered by $I$ for her decision about alternative $A$ and predicate $diff(I, J)$ means that we have two different decision makers $I$ and $J$.

Let us now illustrate the modelling choices above using the ongoing example.

*Example 3.* Given the set of decision makers $\mathcal{C}_\mathcal{I} = \{M, P\}$, let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base with:

- Set of facts $\mathcal{F} = \{\mathcal{J}_{M1}, \mathcal{J}_{M2}, \mathcal{J}_{P1}, \mathcal{J}_{P2}, pushchair(Yoyo), pushchair(Chicco)\}$
- Set of rules $\mathcal{R} = \mathcal{R}_M \cup \mathcal{R}_P$

$$\begin{aligned}\mathcal{R}_P =& \{\forall a, pushchair(a) \wedge \mathcal{J}_{P1} \rightarrow consider(P, a, price^*),\\ & \forall a, pushchair(a) \wedge \mathcal{J}_{P2} \rightarrow consider(P, a, weight^*)\}\\ \mathcal{R}_M =& \{\forall a, pushchair(a) \wedge \mathcal{J}_{M1} \rightarrow consider(M, a, price^*),\\ & \forall a, pushchair(a) \wedge \mathcal{J}_{M2} \rightarrow not\_consider(M, a, weight^*)\}\end{aligned}$$

- Set of negative constraints $\mathcal{N}$ with

$$\begin{aligned}\mathcal{N} =& \{\forall i \forall a \forall c_k, consider(i, a, c_k) \wedge not\_consider(j, a, c_k^*) \wedge pushchair(a)\\ & \wedge diff(i, j) \rightarrow \perp\}\end{aligned}$$

By applying the rules we obtain the set

$$\begin{aligned}\mathcal{F}^* = \mathcal{F} \cup \{& consider(M, Yoyo, price^*),\\ & consider(P, Yoyo, price^*),\\ & consider(P, Yoyo, weight^*),\\ & not\_consider(M, Yoyo, weight^*),\\ & consider(M, Chicco, price^*),\\ & consider(P, Chicco, price^*),\\ & consider(P, Chicco, weight^*),\\ & not\_consider(M, Chicco, weight^*)\}\end{aligned}$$

We can see that the negative constraint is triggered twice. Firstly, $P$ considers the weight for pushchair $Yoyo$ whereas $M$ does not. Secondly, $P$ also considers the weight for pushchair $Chico$ whereas $M$ does not.

### 3.2 Inconsistency regarding the consideration of different alternatives

Each decision maker $I \in \mathcal{C_I}$ gives a set of rules of the form $R_I = \{\forall A \in \mathcal{C}_A\ type(A) \wedge \mathcal{J}_{I_i} \rightarrow alternative(A, I)\}$ which corresponds to the alternatives considered by $I$. Inconsistencies occur when different decision makers do not consider the same alternatives. Therefore negative constraints are of the form $\forall I, J \in \mathcal{C_I} \forall A \in \mathcal{C}_A\ alternative(A, I) \wedge not\_alternative(A, J) \wedge diff(I, J) \rightarrow \perp$. The predicate $not\_alternative(A, I)$ corresponds to the case where alternative $A$ is not considered by $I$ for her decision and predicate $diff(I, J)$ means that we have two different decision makers $I$ and $J$.

*Example 4.* Given the set of decision makers $\mathcal{C_I} = \{M, P\}$, let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base with:

- Set of facts $\mathcal{F} = \{\mathcal{J}_M, \mathcal{J}_P, pushchair(Yoyo), pushchair(Chicco)\}$
- Set of rules $\mathcal{R} = \mathcal{R}_P \cup \mathcal{R}_M$

$$\begin{aligned} \mathcal{R}_P =& \{\forall a, pushchair(a) \wedge \mathcal{J}_P \rightarrow alternative(Yoyo, P) \\ & \wedge not\_alternative(Chicco, P)\} \\ \mathcal{R}_M =& \{\forall a, pushchair(a) \wedge \mathcal{J}_M \rightarrow alternative(Yoyo, M) \\ & \wedge alternative(Chicco, M)\} \end{aligned}$$

- Set of negative constraints $\mathcal{N}$ with

$$\mathcal{N} = \{\forall j \forall i \forall a\ alternative(a, i) \wedge not\_alternative(a, j) \wedge diff(i, j) \rightarrow \perp\}$$

By applying the rules we obtain the set

$$\begin{aligned} \mathcal{F}^* = \mathcal{F} \cup \{& alternative(Yoyo, M), \\ & alternative(Yoyo, P), \\ & alternative(Chicco, M), \\ & not\_alternative(Chicco, P)\} \end{aligned}$$

We can see that $M$ is considering alternative $Chicco$ and $P$ is not considering $Chicco$, so the negative constraint is triggered leading to an inconsistency.

### 3.3 Inconsistency regarding the consideration of different preferences on criteria

Each decision maker $I \in \mathcal{C_I}$ gives a set of rules of the form $\mathcal{R}_I = \{\forall A \in \mathcal{C}_A, \forall C_i^*, C_j^* \in \mathcal{C}_C\ consider(I, A, C_i^*) \wedge consider(I, A, C_j^*) \wedge \mathcal{J}_{I_i} \rightarrow preferred(I, C_i^*, C_j^*)\}$ stating her preferences between criteria $c_i$ and $c_j$. Inconsistencies occur when different decision makers consider different preferences over the criteria. Therefore negative constraints are of the form $\forall A \in \mathcal{C}_A \forall I, J \in \mathcal{C_I} \forall C_i^*, C_j^* \in \mathcal{C}_C, preferred(I, A, C_i^*, C_j^*) \wedge$

$not\_preferred(J, A, C^*_i, C^*_j) \wedge diff(I, J), preferred(I, A, C^*_i, C^*_j) \wedge equivalent(J, A, C^*_i, C^*_j) \wedge diff(I, J), equivalent(I, A, C^*_i, C^*_j) \wedge not\_preferred(J, A, C^*_i, C^*_j) \wedge diff(I, J) \rightarrow \bot$. The atom $not\_preferred(I, A, C^*_i, C^*_j)$ corresponds to the case where criterion $C^*_i$ is not preferred in a pairwise comparison with criterion $C^*_j$ by $I$ for her decision on alternative $A$ and predicate $equivalent(I, A, C^*_i, C^*_j)$ means that criteria $C^*_i$ and $C^*_j$ are equally preferred by $I$.

*Example 5.* Given the set of decision makers $\mathcal{C}_{\mathcal{I}} = \{M, P\}$, let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base with:

- Set of facts $\mathcal{F}$ with

$$\begin{aligned}\mathcal{F} = &\{\mathcal{J}_{M1}, \mathcal{J}_{M2}, \mathcal{J}_{M3}, \mathcal{J}_{P1}, \mathcal{J}_{P2}, \mathcal{J}_{P3}, pushchair(Yoyo),\\ &pushchair(Chicco)\}\end{aligned}$$

- Set of rules $\mathcal{R} = \mathcal{R}_P \cup \mathcal{R}_M$

$$\begin{aligned}\mathcal{R}_P = &\{\forall a, pushchair(a) \wedge \mathcal{J}_{P1} \rightarrow consider(P, a, price^*),\\ &\forall a, pushchair(a) \wedge \mathcal{J}_{P2} \rightarrow consider(P, a, weight^*),\\ &\forall a, consider(P, a, price^*) \wedge consider(P, a, weight^*) \wedge \mathcal{J}_{P3} \rightarrow\\ &not\_preferred(P, a, weight^*, price^*)\}\\ \mathcal{R}_M = &\{\forall a, pushchair(a) \wedge \mathcal{J}_{M1} \rightarrow consider(M, a, price^*),\\ &\forall a, pushchair(a) \wedge \mathcal{J}_{M2} \rightarrow consider(M, a, weight^*),\\ &\forall a, consider(M, a, price^*) \wedge consider(M, a, weight^*) \wedge \mathcal{J}_{M3} \rightarrow\\ &preferred(M, a, weight^*, price^*)\}\end{aligned}$$

- Set of negative constraints $\mathcal{N}$ with

$$\begin{aligned}\mathcal{N} = &\{\forall a \forall i \forall j \forall c_1 \forall c_2, preferred(i, a, c_1, c_2) \wedge not\_preferred(j, a, c_1, c_2) \wedge\\ &diff(i, j) \rightarrow \bot\}\end{aligned}$$

By applying the rules we obtain the set

$$\begin{aligned}\mathcal{F}^* = \mathcal{F} \cup &\{consider(M, Yoyo, price^*),\\ &consider(P, Yoyo, price^*),\\ &consider(P, Yoyo, weight^*),\\ &consider(M, Yoyo, weight^*),\\ &preferred(M, Yoyo, weight^*, price^*),\\ &not\_preferred(P, Yoyo, weight^*, price^*)\\ &consider(M, Chicco, price^*),\\ &consider(P, Chicco, price^*),\\ &consider(P, Chicco, weight^*),\\ &consider(M, Chicco, weight^*),\\ &preferred(M, Chicco, weight^*, price^*),\\ &not\_preferred(P, Chicco, weight^*, price^*)\}\end{aligned}$$

We can see that $M$ and $P$ have different preferences over the criteria ($price^*$ and $weight^*$) for each of the alternatives, i.e., $M$ considers criterion weight more important than price while $P$ considers the opposite. Hence, the negative constraint is triggered leading to an inconsistency.

### 3.4 Inconsistency on the criteria values of the decision makers over the alternatives

Each decision maker $I \in \mathcal{C}_{\mathcal{I}}$ gives a set of rules of the form $\mathcal{R}_I = \{\forall A \in \mathcal{C}_A\, type(A) \wedge \mathcal{J}_{I_i} \rightarrow has\_value(I, A, C^*, V)\}$ that corresponds to the value $V \in \mathcal{C}_V^c$ $I$ gives for criterion $C^*$ for alternative $A$. Inconsistencies occur when different decision makers have different valuation for the same alternative on a specific criterion. The value corresponds to a metric that is related to real data. Although one could think that, since value is objective, it would be the same for all the decision makers, this is not the case in real life simply because not all the decision makers have the same information. Therefore, negative constraints are of the form $\forall I, J \in \mathcal{C}_{\mathcal{I}} \forall A \in \mathcal{C}_A \forall C^* \in \mathcal{C}_C \forall V_1, V_2 \in \mathcal{C}_V^c, has\_value(I, A, C^*, V_1) \wedge has\_value(J, A, C^*, V_2) \wedge type(A) \wedge diff(I, J) \wedge diff(V_1, V_2) \rightarrow \perp$.

*Example 6.* Given the set of decision makers $\mathcal{C}_{\mathcal{I}} = \{M, P\}$, let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base with:

- Set of facts $\mathcal{F} = \{\mathcal{J}_{M1}, \mathcal{J}_{M2}, \mathcal{J}_{P1}, \mathcal{J}_{P2}, pushchair(Yoyo), pushchair(Chicco)\}$
- Set of rules $\mathcal{R} = \mathcal{R}_P \cup \mathcal{R}_M$

$$\begin{aligned}\mathcal{R}_P =& \{\forall a, pushchair(a) \wedge \mathcal{J}_{P1} \rightarrow has\_value(P, Yoyo, price^*, 90),\\ & \forall a, pushchair(a) \wedge \mathcal{J}_{P2} \rightarrow has\_value(P, Chicco, price^*, 80),\\ \mathcal{R}_M =& \{\forall a, pushchair(a) \wedge \mathcal{J}_{M1} \rightarrow has\_value(M, Yoyo, price^*, 70),\\ & \forall a, pushchair(a) \wedge \mathcal{J}_{M2} \rightarrow has\_value(M, Chicco, price^*, 60)\}\end{aligned}$$

- Set of negative constraints $\mathcal{N} = \{\forall I, J \in \mathcal{C}_{\mathcal{I}} \forall A \in \mathcal{C}_A \forall C^* \in \mathcal{C}_C \forall V_1, V_2 \in \mathcal{C}_V^c, has\_value(I, A, C^*, V_1) \wedge has\_value(J, A, C^*, V_2) \wedge type(A) \wedge diff(I, J) \wedge diff(V_1, V_2) \rightarrow \perp\}$

By applying the rules we obtain the set

$$\begin{aligned}\mathcal{F}^* = \mathcal{F} \cup \{& has\_value(M, Yoyo, price^*, 70),\\ & has\_value(P, Yoyo, price^*, 90)\\ & has\_value(M, Chicco, price^*, 60),\\ & has\_value(P, Chicco, price^*, 80\}\end{aligned}$$

We can see that $M$ and $P$ have different values in price criterion for both alternatives (70 and 90 for Yoyo, 60 and 80 for Chicco), so the negative constraint is triggered leading to an inconsistency. Even if the value for price can be objective, inconsistency can occur due to lack of information or different source of information, e.g., different stores can have different prices on the same product.

### 3.5 Inconsistency on the criteria evaluation ratings (preferences) of the decision makers over the alternatives

Each decision maker $I \in \mathcal{C}_{\mathcal{I}}$ gives a set of rules of the form $\mathcal{R}_I = \{\forall A \in \mathcal{C}_A\, type(A) \wedge \mathcal{J}_{I_i} \rightarrow has\_rating(I, A, C^*, R)$ that corresponds to the evaluation rating $R \in \mathcal{C}^c_R$ that $I$ gives for criterion $C^* \in \mathcal{C}_C$ for alternative $A$. Inconsistencies occur when different decision makers have different preferences, i.e., evaluation ratings for the same alternative on a specific criterion. The evaluation rating corresponds to a metric that is related to the preferences of the decision makers. The preferences of the decision makers are subjective and this can be seen through the inconsistencies of the evaluation ratings. Therefore, negative constraints are of the form $\forall I, J \in \mathcal{C}_{\mathcal{I}} \forall A \in \mathcal{C}_A \forall C^* \in \mathcal{C}_C \forall R_1, R_2 \in \mathcal{C}^c_R, has\_rating(I, A, C^*, R_1) \wedge has\_rating(J, A, C^*, R_2) \wedge type(A) \wedge diff(I, J) \wedge diff(R_1, R_2) \rightarrow \perp$..

*Example 7.* In the following example "vfm" stands for the "value for money" criterion. Given the set of decision makers $\mathcal{C}_{\mathcal{I}} = \{M, P\}$, let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base with:

- Set of facts $\mathcal{F} = \{\mathcal{J}_{M1}, \mathcal{J}_{M2}, \mathcal{J}_{P1}, \mathcal{J}_{P2}, pushchair(Yoyo), pushchair(Chicco)\}$
- Set of rules $\mathcal{R} = \mathcal{R}_P \cup \mathcal{R}_M$

$$\begin{aligned}\mathcal{R}_P =&\{\forall a, pushchair(a) \wedge \mathcal{J}_{P1} \rightarrow has\_rating(P, Yoyo, vfm^*, 9),\\ &\forall a, pushchair(a) \wedge \mathcal{J}_{P2} \rightarrow has\_rating(P, Chicco, vfm^*, 8),\\ \mathcal{R}_M =&\{\forall a, pushchair(a) \wedge \mathcal{J}_{M1} \rightarrow has\_rating(M, Yoyo, vfm^*, 7),\\ &\forall a, pushchair(a) \wedge \mathcal{J}_{M2} \rightarrow has\_rating(M, Chicco, vfm^*, 7)\}\end{aligned}$$

- Set of negative constraints $\mathcal{N}$ with

$$\begin{aligned}\mathcal{N} =&\{\forall i \forall a \forall c_k \forall v_n \forall v_{n\prime}, n \in [1 \ldots N], has\_rating(i, a, c_k, v_n)\\ &\wedge has\_rating(j, a, c_k, v_{n\prime}) \wedge pushchair(a) \wedge diff(i, j) \wedge\\ &\exists n, n\prime \mid diff(v_n, v_{n\prime}) \rightarrow \perp\}\end{aligned}$$

By applying the rules we obtain the set

$$\begin{aligned}\mathcal{F}^* = \mathcal{F} \cup \{&has\_rating(M, Yoyo, vfm^*, 7),\\ &has\_rating(P, Yoyo, vfm^*, 9)\\ &has\_rating(M, Chicco, vfm^*, 7),\\ &has\_rating(P, Chicco, vfm^*, 8\}\end{aligned}$$

The obtained facts show the ratings given by $P$ and $M$ for the vfm of $Yoyo$ and $Chicco$. We can see that they have different perspective on judging the "vfm" criterion for both alternatives ($M$ gives 7 and $P$ gives 9 for Yoyo, while they give 7 and 8 for Chicco), so the negative constraint is triggered leading to an inconsistency.

## 4 Discussion and related work

In this work, we proposed a framework to model the multi-criteria decision problem as reasoning in presence of inconsistency problem. We used existential rules for the modelling language of the knowledge base and employed repair techniques for reasoning.

Please note that existing work considers alternative inconsistency proof semantics. In knowledge representation and reasoning argumentation is also a well known method for handling inconsistency. Several researchers have proposed the use of argumentation in decision-making. The work of Fox and Parsons [17] is among the first argumentative approaches to decision-making by clarifying the difference between argumentation for actions and argumentation for beliefs. Most of the argumentative approaches ([18], [21]) aim to select the best option but certain approaches (such as those by Amgoud and Prade [2]) use a two step process that allows to further analyse the different options presented to the user. Using argumentation techniques for reasoning with existential rules knowledge bases was proven to yield semantically equivalent results as to repair based techniques. Furthermore, argumentation has been long investigated for its principled human computer interaction advantages (see for instance the results of using argumentation for handling inconsistency in existential rules of [4, 6, 5]). However, in this context, argumentation is computationally extensive process that involves a large overhead due to the computation of all arguments and attacks over a knowledge base.

## Acknowledgements.

The work of Nikos Karanikolas was supported with a scholarship from IKY funded by the action “Support of Postdoctoral Researchers” from the resources of the EP “Human Resources Development, Education and Lifelong Learning” with priority axes $6, 8, 9$ and is and co-funded by the European Social Fund - ESF and the Greek state.

Part of this work has been carried out while Nikos Karanikolas was employed by INRA, France. The authors acknowledge the support of the H2020 NoAW project.

The authors are grateful to Matéi Rabatel for providing the real world use-case.

## References

1. S. Abiteboul, R. Hull, and V. Vianu. *Foundations of databases*. Addison-Wesley Reading, 1995.
2. L. Amgoud and H. Prade. Using arguments for making and explaining decisions. *Artificial Intelligence*, 173(3–4):413–436, 2009.
3. M. Arenas, L. Bertossi, and J. Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 68–79. ACM, 1999.
4. A. Arioua and M. Croitoru. Dalek: a tool for dialectical explanations in inconsistent knowledge bases. In *Proceedings of the 6th International Conference on Computational Models of Argument, COMMA 2016*, 2016.
5. A. Arioua and M. Croitoru. Dialectical characterization of consistent query explanation with existential rules. In *Proceedings of the 29th International Florida Artificial Intelligence Research Society Conference (FLAIRS’16)*, pages 14–19, 2016.

6. A. Arioua, M. Croitoru, L. Papaleo, N. Pernelle, and S. Rocher. On the explanation of sameas statements using argumentation. In *Proceedings of the 10th International Conference on Scalable Uncertainty Management (SUM'16)*, 2016.
7. J. Baget, S. Benferhat, Z. Bouraoui, M. Croitoru, M. Mugnier, O. Papini, S. Rocher, and K. Tabia. Inconsistency-tolerant query answering: Rationality properties and computational complexity analysis. In L. Michael and A. C. Kakas, editors, *Logics in Artificial Intelligence - 15th European Conference, JELIA 2016, Larnaca, Cyprus, November 9-11, 2016, Proceedings*, volume 10021 of *Lecture Notes in Computer Science*, pages 64–80, 2016.
8. J.-F. Baget, M. Leclère, M.-L. Mugnier, and E. Salvat. On rules with existential variables: Walking the decidability line. *Artificial Intelligence*, 175(9–10):1620–1654, 2011.
9. J.-F. Baget, M.-L. Mugnier, S. Rudolph, and M. Thomazo. Walking the complexity lines for generalized guarded existential rules. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 712–717, 2011.
10. C. Beeri and M. Y. Vardi. The implication problem for data dependencies. In *International Colloquium on Automata, Languages, and Programming*, pages 73–85. Springer, 1981.
11. R. Benayoun, B. Roy, and B. Sussman. ELECTRE: une méthode pour guider le choix en présence des points de vue multiples. Technical report, SEMA-METRA International, Direction Scientifique, 1966. Note de travail 49.
12. M. Bienvenu and R. Rosati. Tractable approximations of consistent query answering for robust ontology-based data access. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, 2013.
13. A. Calì, G. Gottlob, and T. Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. *Journal of Web Semantics*, 14:57–83, 2012.
14. M. Chein and M. Mugnier. *Graph-based Knowledge Representation - Computational Foundations of Conceptual Graphs*. Advanced Information and Knowledge Processing. Springer, 2009.
15. J. Chomicki. Consistent query answering: Five easy pieces. In *International Conference on Database Theory (ICDT'07)*, pages 1–17. Springer, 2007.
16. R. Fagin. *Encyclopedia of Database Systems*, chapter Tuple-Generating Dependencies, pages 3201–3202. Springer US, 2009.
17. J. Fox and S. Parsons. On using arguments for reasoning about actions and values. In *Proceedings of the AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning*, pages 55–63. AAAI Press, 1997.
18. N. I. Karacapilidis and D. Papadias. A group decision and negotiation support system for argumentation based reasoning. In *Selected Papers from the Workshop on Reasoning with Incomplete and Changing Information and on Inducing Complex Representations(PRICAI '96)*, pages 188–205, 1998.
19. D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi, and D. F. Savo. Inconsistency-tolerant semantics for description logics. In *Proceedings of the International Conference on Web Reasoning and Rule Systems (RR'10)*, pages 103–117. Springer-Verlag, 2010.
20. J. Masthoff. Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction*, 14(1):37–85, 2004.
21. M. Morge and P. Mancarella. The hedgehog and the fox. an argumentation-based decision support system. In *Proceedings of the 4th International Workshop on Argumentation in Multi-Agent Systems (ArgMas'07)*, pages 55–68, 2007.
22. A. Ostanello. *Outranking Methods*, pages 41–60. Springer Berlin Heidelberg, Berlin, Heidelberg, 1985.
23. N. Rescher and R. Manor. On inference from inconsistent premisses. *Theory and decision*, 1(2):179–217, 1970.