Fundamenta Informaticae XX (2011) 1–21
IOS Press

A Possibilistic Argumentation Decision Making Framework with Default Reasoning

Juan Carlos Nieves
Universitat Politècnica de Catalunya
Dept. Llenguatges i Sistemes Informàtics
jcnieves@lsi.upc.edu

Roberto Confalonieri
Universitat Politècnica de Catalunya
Dept. Llenguatges i Sistemes Informàtics
confalonieri@lsi.upc.edu

Abstract. In this paper, we introduce a possibilistic argumentation-based decision making framework which is able to capture uncertain information and exceptions/defaults. In particular, we define the concept of a possibilistic decision making framework which is based on a possibilistic default theory, a set of decisions and a set of prioritized goals. This set of goals captures user preferences related to the achievement of a particular state in a decision making problem. By considering the inference of the possibilistic well-founded semantics, the concept of argument with respect to a decision is defined. This argument captures the feasibility of reaching a goal by applying a decision in a given context. The inference in the argumentation decision making framework is based on basic argumentation semantics. Since some basic argumentation semantics can infer more than one possible scenario of a possibilistic decision making problem, we define some criteria for selecting potential solutions of the problem.

Keywords: Argumentation Theory, Well-Founded Semantics, Decision Making

1. Introduction

Decision making can be regarded as a reasoning process for selecting an action among several alternatives. Any decision problem looks for the best action(s) for pursuing a goal. In our common life, decision

Address for correspondence: Universitat Politècnica de Catalunya, Dept. Llenguatges i Sistemes Informàtics, Edifici K2M, Despatx 201, C/ Jordi Girona Salgado 1-3, E - 08034 Barcelona, {jcnieves,confalonieri}@lsi.upc.edu.

making processes are present in any moment and most of them have to deal with uncertainty. In fact, any decisions that we make in our common life are based on beliefs concerning the likelihood of uncertain events. In fact, we commonly use arguments such as "I think that ...", "chances are ...", "it is *probable* that ...", "it is *plausible* that ...", *etc.*, for supporting our decisions. In these kinds of arguments, we usually appeal to our experience or our common sense. It is not surprising to think that a reasoning based on these kinds of statements could reach *biased conclusions*. However, these conclusions could reflect the experience or common sense of an expert. Pelletier and Elio pointed out in [34] that people simply have tendencies to ignore certain information because of the (evolutionary) necessity to make decisions quickly. This *necessity* gives rise to *biases* in judgments concerning what they *really* want to do.

From the common sense reasoning perspective, the decision making approaches based on cognitive-states arise as flexible formalisms for supporting decision making based on beliefs of the world [38]. In particular, approaches based on argumentation theory [5, 22, 23, 37] seem to be flexible enough for dealing with the different forms of information for justifying/explaining rational decisions. Indeed, from a practical perspective, argumentation theory provides a versatile computational model for developing advanced services for decision support and for computer human dialogues in which explanation and rationale play a central role [23].

As far as we know, there are few attempts to formalize argument-based decision making under uncertainty. Some of the most representative approaches are

- Bonet and Geffner's approach [11],
- works based on Logic of Argumentation [24, 29] and more recently,
- works based on the Possibilistic Logic [1, 2, 5].

These approaches offer different features in the process of expressing a decision making problem. However, some of these approaches have a quite limited specification language for expressing *exceptions* (also called *defaults* in non-monotonic reasoning). The specification of exceptions in the representation of a decision making problem takes relevance in domains in which the available information is incomplete [7]. On the other hand, the identification of exceptions of a given argument characterizes a relation of *undercut* between arguments which is not based on strong negation [21].

The use of logic specification languages with *negation as failure* is a successful approach for encoding knowledge with exceptions/defaults. In the last two decades, one of the most successful logic programming approaches has been Answer Set Programming (ASP) [7]. ASP is the realization of much theoretical work on Non-monotonic Reasoning and Artificial Intelligence applications. It represents a new paradigm for logic programming that allows, using the concept of *negation as failure*, the handling of problems with *default knowledge* and the production of non-monotonic reasoning. ASP usually is based on the so called *Stable Model Semantics* [27]. Another prominent logic programming semantics in the context of ASP is the so called *Well-Founded Semantics (WFS)* [26]. Unlike the stable model semantics which follows a credulous reasoning approach, WFS performs a skeptical reasoning approach. WFS can be regarded as an approximation of the stable model semantics. Dix in [16] showed that WFS satisfies a good set of expected properties in the context of non-monotonic reasoning.

In [33], a possibilistic framework for reasoning under uncertainty was proposed. This framework is a combination of WFS and possibilistic logic [19]. Possibilistic logic is based on possibilistic theory, in which, at the mathematical level, degrees of possibility and necessity are closely related to fuzzy sets

[19]. Due to the natural properties of possibilistic logic and WFS, this approach allows us to deal with a reasoning that is at the same time *non-monotonic* and *uncertain*. This possibilistic version of WFS has some nice features: it can be regarded as an approximation of the possibilistic stable model semantics [30, 31] and it is polynomial time computable. Another distinguishing characteristic of the approach presented in [33] is that it makes possible the use of explicit labels such as *certain*, *probable*, *plausible*, *etc*., in order to capture the incomplete state of a belief in a normal logic program. All these features of the possibilistic WFS define a potential platform for defining an argument-based decision making under uncertainty which can deal with default reasoning.

In this paper, we introduce a possibilistic argumentation-based decision making framework which is able to capture uncertain information and exceptions/defaults. In particular, we define the concept of a possibilistic decision making framework which is based on a possibilistic default theory, a set of decisions and a set of prioritized goals. This set of goals captures user preferences related to the achievement of a particular state in a decision making problem. By considering the inference of the possibilistic well-founded semantics, the concept of argument with respect to a decision is defined. This argument captures the feasibility of reaching a goal by applying a decision in a given context. Since the possibilistic well-founded semantics is a three valued semantics, the relation of attacks between arguments is defined in terms of complementary atoms and assumptions. The inference in the argumentation decision making framework is based on basic argumentation semantics (following Dung's style [21]). Since some basic argumentation semantics can infer more than one possible scenario of a possibilistic decision making problem, we define some criteria for selecting potential solutions to the problem.

The paper is organized as follows: after presenting the relevant concepts we will use throughout the paper (Section 2), in Section 3, we introduce our proposal for a possibilistic argumentation decision making framework for dealing with incomplete knowledge. In Section 4, some criteria for comparing arguments and set of arguments are defined. In Section 5, we compare our work with other noticeable approaches in the literature. Finally, Section 6 concludes the paper discussing the achieved results and pointing out some future work. Along the paper we use a running example to exemplify our approach.

2. Background

In this section, we introduce the necessary terminology in order to have a self-contained document. We assume that the reader has familiarity with basic concepts of classic logic and logic programming.

We start by introducing the syntax and a semantics for normal logic programs. After that, the syntax and the semantics are extended for the setting of a possibilistic logic programming framework.

2.1. Non-Possibilistic Normal Logic Programs

The language of propositional logic has an alphabet consisting of

(i) propositional symbols: $p_0, p_1, ...$

(ii) connectives : $\vee, \wedge, \leftarrow, \neg, \ not, \top$

(iii) auxiliary symbols : (,).

in which $\vee, \wedge, \leftarrow$ are 2-place connectives, $\neg$, not are 1-place connectives and $\top$ is a 0-place connective. The propositional symbols, $\top$, and the propositional symbols of the form $\neg p_i$ ($i \geq 0$) stand for the indecomposable propositions, which we call *atoms*, or *atomic propositions*. Atoms negated by $\neg$ will be called *extended atoms*. We will use the concept of atom without paying attention to whether it is an extended atom or not. The negation sign $\neg$ is regarded as the so called *strong negation* by the ASP's literature and the negation not as the *negation as failure*. A literal is an atom, a (called positive literal), or the negation of an atom $not\ a$ (called negative literal). Given a set of atoms $\{a_1, ..., a_n\}$, we write $not\ \{a_1, ..., a_n\}$ to denote the set of literals $\{not\ a_1, ..., not\ a_n\}$.

An extended normal clause, *C*, is denoted:

$$a \leftarrow b_1, \ldots, b_j, not\ b_{j+1}, \ldots, not\ b_{j+n} \tag{1}$$

where $j + n \geq 0$, a is an atom and each b_i ($1 \leq i \leq j + n$) is an atom. When $j + n = 0$ the clause is an abbreviation of $a \leftarrow \top$ such that $\top$ is the propositional symbol that always evaluates to true. In a slight abuse of notation, we sometimes write the clause 1 as $a \leftarrow \mathcal{B}^+, not\ \mathcal{B}^-$, where $\mathcal{B}^+ := \{b_1, \ldots, b_j\}$ and $\mathcal{B}^- := \{b_{j+1}, \ldots, b_{j+n}\}$. An extended normal program P is a finite set of extended normal clauses. When $n = 0$, the clause is called *extended definite clause*. *An extended definite logic program* is a finite set of extended definite clauses. By $\mathcal{L}_P$, we denote the set of atoms in the signature of P. Let $Prog_{\mathcal{L}}$ be the set of all normal programs with atoms from $\mathcal{L}$.

We will manage the strong negation ($\neg$) in our logic programs as it is done in ASP [7]. Basically, each atom of the form $\neg a$ is replaced by a new atom symbol a' which does not appear in the language of the program. For instance, let P be the extended normal program:

$$a \leftarrow q. \qquad \neg q \leftarrow r. \qquad q \leftarrow \top. \qquad r \leftarrow \top.$$

Then replacing the atom $\neg q$ with a new atom symbol q', we will have:

$$a \leftarrow q. \qquad q' \leftarrow r. \qquad q \leftarrow \top. \qquad r \leftarrow \top.$$

In order not to allow inconsistent models from logic programs, a normal clause of the form $f \leftarrow q, q', f$ such that $f \notin \mathcal{L}_P$ is added.

2.2. Well-Founded Semantics

In this section, we present a standard definition of the well-founded semantics in terms of rewriting systems. We start by presenting a definition w.r.t. a 3-valued logic semantics.

Definition 2.1. (SEM)
[18] For a normal logic program P, we define HEAD$(P) := \{a \mid a \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^- \in P\}$ — the set of all head-atoms of P. We also define SEM$(P) = \langle P^{true}, P^{false} \rangle$, where $P^{true} := \{p \mid p \leftarrow \top \in P\}$ and $P^{false} := \{p \mid p \in \mathcal{L}_P \backslash \text{HEAD}(P)\}$. SEM$(P)$ is also called a model of P.

In order to present a characterization of the well-funded semantics in terms of rewriting systems, we define some basic transformation rules for normal logic programs.

Definition 2.2. (Basic Transformation Rules)
[18] A transformation rule is a binary relation on $\text{Prog}_{\mathcal{L}}$. The following transformation rules are called *basic*. Given a program $P \in \text{Prog}_{\mathcal{L}}$ we define:

RED$^+$: This transformation can be applied to P, if there is an atom a which does not occur in HEAD(P). **RED$^+$** transforms P to the program where all occurrences of $not\ a$ are removed.

RED$^-$: This transformation can be applied to P, if there is a rule $a \leftarrow \top \in P$. **RED$^-$** transforms P to the program where all clauses that contain $not\ a$ in their bodies are deleted.

Success: Suppose that P includes a fact $a \leftarrow \top$ and a clause $q \leftarrow body$ such that $a \in body$. Then we replace the clause $q \leftarrow body$ by $q \leftarrow body \setminus \{a\}$.

Failure: Suppose that P contains a clause $q \leftarrow body$ such that $a \in body$ and $a \notin HEAD(P)$. Then we erase the given clause.

Loop: We say that P_2 results from P_1 by $Loop_A$ if, by definition, there is a set A of atoms such that:

1. for each rule $a \leftarrow body \in P_1$, if $a \in A$, then $body \cap A \neq \emptyset$,
2. $P_2 := \{a \leftarrow body \in P_1 | body \cap A = \emptyset\}$,
3. $P_1 \neq P_2$.

Let CS_0 be the rewriting system containing the basic transformation rules: **RED$^+$**, **RED$^-$**, **Success**, **Failure**, and **Loop**.

We denote the uniquely determined normal form of a program P with respect to the rewriting system CS_0 by $norm_{CS_0}(P)$. CS_0 induces a semantics SEM_{CS_0} as follows:

$$\text{SEM}_{CS_0}(P) := \text{SEM}(norm_{CS_0}(P))$$

In order to illustrate the basic transformation rules, let us consider the following example.

Example 2.1. Let P be the following normal logic program:

$$b \leftarrow not\ a. \qquad c \leftarrow not\ b. \qquad c \leftarrow a.$$

Now, let us apply CS_0 to P. Since $a \notin HEAD(P)$, then we can apply **RED$^+$** to P. Thus we get:

$$b \leftarrow \top. \qquad c \leftarrow not\ b. \qquad c \leftarrow a.$$

Observe that now we can apply **RED$^-$** to the new program, thus we get:

$$b \leftarrow \top. \qquad\qquad c \leftarrow a.$$

Finally, we can apply **Failure** to the new program, thus we get:

$$b \leftarrow \top.$$

This last program is called the *normal form* of P w.r.t. CS_0, because none of the transformation rules from CS_0 can be applied.

WFS was introduced in [26] and it was characterized in terms of rewriting systems in [13]. This characterization is defined as follows:

Lemma 2.1. [13] CS_0 is a confluent rewriting system. It induces a 3-valued semantics that is the well-founded semantics.

Given a normal logic program P, by $WFS(P)$, we denote the well-founded model of P.

Example 2.2. Let P be the normal logic program introduced in Example 2.1. As we saw in Example 2.1, $norm_{CS_0}(P)$ is:

$$b \leftarrow \top.$$

This means that $WFS(P) = \langle \{b\}, \{a, c\} \rangle$

2.3. Possibilistic Logic Programs

In this section, we extend the syntax of extended normal logic programs in order to capture uncertain information. To this end, we start introducing some basic concepts in terms of possibilistic logic programs. These concepts were first introduced in [30].

A *possibilistic atom* is a pair $p = (a, q) \in \mathcal{A} \times Q$, where $\mathcal{A}$ is a finite set of atoms and $(Q, \leq)$ is a lattice[1]. We apply the projection $*$ over p as follows: $p^* = a$. Given a set of possibilistic atoms S, we define the generalization of $*$ over S as follows: $S^* := \{p^* | p \in S\}$. Given a lattice $(Q, \leq)$ and $S \subseteq Q$, $LUB(S)$ denotes the least upper bound of S and $GLB(S)$ denotes the greatest lower bound of S.

We define the syntax of a valid extended possibilistic normal logic program as follows. Let $(Q, \leq)$ be a lattice. An extended possibilistic normal clause r is of the form:

$$\alpha : a \leftarrow \mathcal{B}^+, \ not \ \mathcal{B}^- \tag{2}$$

where $\alpha \in Q$. The projection $*$ over the possibilistic clause r is: $r^* = a \leftarrow \mathcal{B}^+, \ not \ \mathcal{B}^-$. $n(r) = \alpha$ is a necessity degree representing the certainty level of the information described by r.

An extended possibilistic normal logic program P is a tuple of the form $\langle (Q, \leq), N \rangle$, where $(Q, \leq)$ is a lattice and N is a finite set of extended possibilistic normal clauses. The generalization of the projection $*$ over P is as follows: $P^* := \{r^* | r \in N\}$. Observe that P^* is an extended logic normal program. When P^* is an extended definite program, P is called an extended possibilistic definite logic program.

Given an extended possibilistic normal logic program $P = \langle (\mathcal{Q}, \leq), N \rangle$, we define the *$\alpha$-cut* and the *strict α-cut* of P, denoted respectively by P_α and $P_{\overline{\alpha}}$, by

$P_\alpha = \langle (\mathcal{Q}, \leq), N_\alpha \rangle$ such that $N_\alpha := \{r | r \in N \text{ and } n(r) \geq \alpha\}$
$P_{\overline{\alpha}} = \langle (\mathcal{Q}, \leq), N_{\overline{\alpha}} \rangle$ such that $N_{\overline{\alpha}} := \{r | r \in N \text{ and } n(r) > \alpha\}$

In order to illustrate the expressiveness of the class of possibilistic normal logic programs, we introduce the following example.

[1] In this paper we assume that the lattice is finite.

Example 2.3. Let us consider Savage's rotten egg decision problem [36] according to which: *An agent is preparing an omelette. 5 fresh eggs are already in the omelette. There is one more egg. The egg either is fresh or rotten.* The status of the egg is *uncertain.*

- The agent can

  1. add it to the omelette which means the whole omelette may be wasted,
  2. throw it away, which means one egg may be wasted, or
  3. put it in a cup, check whether it is ok or not and put it into the omelette in the former case, throw it away in the latter. In any case, a cup has to be washed.

- the agent prefers

  – a six-egg omelette and a cup to wash, over one with five eggs and not to wash a cup over having nothing to eat

In the example, decisions correspond to actions and states correspond to the possible states of the egg, *i.e.* decisions $C := \{inOmelette, inCup, throwAway\}$, and states $S := \{fresh, rotten\}$. The decision problem can be modeled by means of the following logic program:[2]

$$P := \left\{ \begin{array}{cc} \mathbf{1} : inO \leftarrow not\ inC, not\ tA. & \mathbf{1} : 5o \leftarrow tA. \\ \mathbf{1} : inC \leftarrow not\ inO, not\ tA. & \mathbf{1} : 6o \leftarrow fr, inO. \\ \mathbf{1} : tA \leftarrow not\ inC, not\ inO. & \mathbf{1} : 0o \leftarrow ro, inO. \\ \lambda : ro \leftarrow not\ fr. & \mathbf{1} : 6o \leftarrow fr, inC. \\ \gamma : fr \leftarrow not\ ro. & \mathbf{1} : 5o \leftarrow ro, inC. \\ \mathbf{1} : w \leftarrow inC. & \mathbf{1} : \neg w \leftarrow not\ inC. \end{array} \right\}$$

where $0 < \lambda \leq 1$, and $0 < \gamma \leq 1$.

This decision making problem has different possible solutions. Then, some decision strategies can be defined taking into account the preferences of the agent. For instance, in [14], preferences are modeled by means of the ordered disjunction connective $\times$ and a preference order between the solution is obtained. However, in that model, no uncertainty about the state of the eggs can be taken into account, since all states that are not negligible (*i.e.* excluded answer sets) are considered plausible. As such, the decision of the best decision is based on empirical comparison criteria only (see [14] for details). In the following, we propose an argumentation approach for making the best decision which considers the certainty degrees of the plausible states. In the argumentation framework, we will construct arguments for supporting each decision and we will show how a decision can be made considering uncertainty and preference degrees of the arguments built. Since the solution to the problem depends on the problem formulation and on the uncertainty in it, we do not expect to give a general solution of the problem. Nevertheless, some potential solutions to the problem are outlined and discussed.

[2] In this program, we assume the lattice $\langle \mathcal{Q}, \leq \rangle$ such that $\mathcal{Q} = (0, 1]$ and $\leq$ is the classical relation between rational numbers.

2.4. Possibilistic Well-Founded Semantics

In this section, the possibilistic Well-Founded Semantics is introduced. This definition is presented in two steps: first, for the class of possibilistic definite logic programs and, after that, the definition is extended for the class of possibilistic normal logic programs.

2.4.1. Extended Possibilistic Definite Logic Programs

In this subsection, we are going to deal with the class of extended possibilistic definite logic programs. We start by introducing the fix-point operator ΠCn which is a fundamental concept for inferring the possibilistic well-founded model.

In order to define the fix-point operator ΠCn, let us introduce some basic definitions. Given a possibilistic logic program P and $x \in \mathcal{L}_{P^*}$, $H(P, x) := \{r \in P | head(r^*) = x\}$.

Definition 2.3. [33] Let $P = \langle (Q, \leq), N \rangle$ be a possibilistic definite logic program, $r \in N$ such that r is of the form $\alpha : a \leftarrow b_1, \ldots, b_n$ and A be a set of possibilistic atoms,

- r is β-applicable in A with $\beta := GLB\{\alpha, \alpha_1, \ldots, \alpha_n\}$ if $\{(b_1, \alpha_1), \ldots, (b_n, \alpha_n)\} \subseteq A$.

- r is $\perp_Q$-applicable otherwise.

And then, for all atom $x \in \mathcal{L}_{P^*}$ we define:

$$App(P, A, x) := \{r \in H(P, x) | r \text{ is } \beta\text{-applicable in } A \text{ and } \beta > \perp_Q\}$$

$\perp_Q$ denotes the bottom element of Q.

Observe that this definition is based on the inferences rules of possibilistic logic. Now, we introduce an operator which is based on Definition 2.3.

Definition 2.4. [33] Let P be a possibilistic definite logic program and A be a set of possibilistic atoms. The immediate possibilistic consequence operator ΠT_P maps a set of possibilistic atoms to another one by this way:

$$\begin{aligned} \Pi T_P(A) := \quad & \{(x, \delta) | x \in HEAD(P^*), App(P, A, x) \neq \emptyset, \\ & \quad \delta = LUB_{r \in App(P,A,x)}\{\beta | r \text{ is } \beta\text{-applicable in } A\}\} \end{aligned}$$

Then, the iterated operator ΠT_P^k is defined by

$$\Pi T_P^0 = \emptyset \text{ and } \Pi T_P^{n+1} = \Pi T_P(\Pi T_P^n), \forall n \geq 0$$

Observe that ΠT_P is a monotonic operator; therefore, ΠT_P always reaches a fix-point which we call the set of possibilistic consequences of P and we denote it by $\Pi Cn(P)$.

By considering the operator ΠCn, we define the possibilistic well-founded semantics for extended possibilistic definite logic program as follows: Let $(Q, \leq)$ be a lattice such that $\top_Q$ is the top-element of Q and S be a set of atoms, then $Q_{\top_Q}(S) := \{(a, \top_Q) | a \in S\}$.

Definition 2.5. [33] Let P be an extended possibilistic definite logic program. S_1 be a set of possibilistic atoms, S_2 be a set of atoms such that $\langle S_1^*, S_2\rangle$ is the well-founded model of P^*. $\langle S_1, Q_{\top_Q}(S_2)\rangle$ is the possibilistic well-founded model of P if and only if $S_1 = \Pi Cn(P)$.

2.4.2. Extended Possibilistic Normal Programs

In this subsection, we are going to deal with the class of extended possibilistic normal programs. To this end, we define a single reduction of an extended possibilistic normal logic program w.r.t. a set of atoms. This reduction is defined as follows:

Definition 2.6. [33] Let P be an extended possibilistic logic program and S be a set of atoms. We define $R(P, S)$ as the extended possibilistic logic program obtained from P by deleting

i) all the formulae of the form $not\ a$ in the bodies of the possibilistic clauses such that $a \in S$, and

ii) each possibilistic clause that has a formula of the form $not\ a$ in its body.

Observe that $R(P, S)$ does not have negative literals. This means that $R(P, S)$ is an extended possibilistic definite logic program.

By considering the fix-point operator $\Pi Cn(P)$ and the reduction $R(P, A)$, the possibilistic version of the well-founded semantics for extended possibilistic normal logic programs is defined as follows:

Definition 2.7. (Possibilistic Well-founded Semantics)
[33] Let $P = \langle (Q, \leq), N\rangle$ be an extended possibilistic normal logic program, S_1 be a set of possibilistic atoms, S_2 be a set of atoms such that $\langle S_1^*, S_2\rangle$ is the well-founded model of P^*. $\langle S_1, Q_{\top_Q}(S_2)\rangle$ is the possibilistic well-founded model of P if and only if $S_1 = \Pi Cn(R(P, S_2))$. By $P\_WFS(P)$, we denote the possibilistic well-founded model of P.

3. Possibilistic Argumentation-based Decision Framework

In this section, we introduce our main results, *i.e.* a framework for capturing a possibilistic decision making problem by means of a possibilistic argumentation-based inference.

Generally speaking, a possibilistic decision making problem follows a structure of cognitive states, namely beliefs, desires and intentions. In fact, the beliefs that an agent has about the world are captured by a possibilistic knowledge base, while intentions and goals of the given agent are expressed in terms of a set of decisions and a set of prioritized goals. Therefore, we can define:

Definition 3.1. A possibilistic decision making framework is a tuple $\langle P, \mathcal{D}, \mathcal{G}\rangle$ in which:

- P is a possibilistic normal logic program.
- $\mathcal{D} = \{d_1, \ldots, d_n\}$ is a set of decision atoms such that $\mathcal{D} \subseteq \mathcal{L}_{P^*}$, $\mathcal{D}^*$ denotes the set of all possible decisions.
- $\mathcal{G} = \{(g_1, \beta_1), \ldots, (g_m, \beta_m)\}$ is a set of possibilistic atoms such that $\mathcal{G}^* \subseteq \mathcal{L}_{P^*}$, $\mathcal{G}^*$ denotes the set of all possible goals and $\beta_j (1 \leq j \leq n)$ represents the priority of the goal g_j.

- $\mathcal{D}^* \cap \mathcal{G}^* = \emptyset$.

This structure of a decision making framework was already explored under the inference of possibilistic logic in [5]. Unlike the possibilistic decision making framework presented in [5], which is restricted to theories of possibilistic logic, the possibilistic decision making framework of Definition 3.1 is based on a possibilistic theory with *negation as failure*. Indeed, the user is able to express assumptions by means of negation as failure.

In order to illustrate the concept of possibilistic decision making framework, let us consider the following example.

Example 3.1. Let us consider the knowledge described by the program in Example 2.3 with certainty values associate with each program rule. Then, we can define:

$$P := \left\{ \begin{array}{cc} \lambda : ro \leftarrow not\ fr. & \mathbf{1} : 6o \leftarrow fr, inC. \\ \gamma : fr \leftarrow not\ ro. & \mathbf{1} : 5o \leftarrow ro, inC. \\ \mathbf{1} : 0o \leftarrow ro, inO. & \mathbf{1} : \neg w \leftarrow not\ inC \\ \mathbf{1} : 5o \leftarrow tA. & \mathbf{1} : 6o \leftarrow fr, inO. \\ \mathbf{1} : w \leftarrow inC. & \end{array} \right\} \qquad \mathcal{G} := \left\{ \begin{array}{ll} (6o, 1), & (w, 1), \\ (5o, \beta), & (\neg w, \beta), \\ (0o, \delta), & \end{array} \right\}$$

$$\mathcal{D} := \left\{ inC, \quad inO, \quad tA \right\}$$

where $1 > \beta > \delta$. The goals are represented by prioritized atoms. Please observe how the priority assigned to the goals reflects the preference ordering of the agent in Example 2.3.

Once we have defined a structure for capturing a decision-making problem, we can define an argumentation decision-making inference. As usual in an argumentation inference, our argumentation decision-making inference consists of four steps:

1. Construction of arguments;
2. Definition of relationships between arguments;
3. Definitions of status of arguments;
4. Selection of decisions.

Since our possibilistic decision making framework is based on a possibilistic default theory, a possibilistic default reasoning inference for building arguments is required. For this purpose, we consider the possibilistic version of the well-founded semantics (Definition 2.7).

Definition 3.2. Let $F = \langle P, \mathcal{D}, \mathcal{G} \rangle$ be a possibilistic decision making framework. An argument of a decision $d \in \mathcal{D}$ is a tuple $A = \langle S, d, (g, \alpha) \rangle$ such that:

1. $(g, \alpha) \in T$ and $g \in \mathcal{G}$ such that $P\_WFS(S \cup \{1 : d \leftarrow \top.\}) = \langle T, F \rangle$.
2. $S \subseteq P$ such that S is a minimal set ($\subseteq$) among the subsets of P satisfying 1.

By $\mathcal{A}_F$, we denote the set of all arguments built from F.

Observe that an argument is capturing the feasibility of reaching a goal by applying a decision in a given state of the world. On the other hand, an argument by itself characterizes a justification/explanation of a decision in a given context for reaching a goal. It is worth mentioning that the possibilistic well-founded semantics is polynomial time computable; hence, the process of building arguments is quite efficient.

In order to illustrate the construction of arguments, let us consider the following example.

Example 3.2. Following Definition 3.2 and the possibilistic decision making framework introduced in Example 3.1, the following arguments can be obtained:

$$\begin{array}{llll} A_1 = & \langle\{\mathbf{1} : 6o \leftarrow fr, inO. & A_2 = & \langle\{\mathbf{1} : 6o \leftarrow fr, inC. \\ & \gamma : fr \leftarrow not\ ro.\}, & & \gamma : fr \leftarrow not\ ro.\}, \\ & inO, & & inC, \\ & (6o, \gamma)\rangle & & (6o, \gamma)\rangle \end{array}$$

$$\begin{array}{llll} A_3 = & \langle\{\mathbf{1} : w \leftarrow inC.\}, & A_4 = & \langle\{\mathbf{1} : 5o \leftarrow tA.\}, \\ & inC, & & tA, \\ & (w, 1)\rangle & & (5o, 1)\rangle \end{array}$$

$$\begin{array}{llll} A_5 = & \langle\{\mathbf{1} : 5o \leftarrow ro, inC. & A_6 = & \langle\{\mathbf{1} : \neg w \leftarrow not\ inC.\}, \\ & \lambda : ro \leftarrow not\ fr.\}, & & tA, \\ & inC, & & (\neg w, 1)\rangle \\ & (5o, \lambda)\rangle & & \end{array}$$

$$\begin{array}{llll} A_7 = & \langle\{\mathbf{1} : \neg w \leftarrow not\ inC.\} & A_8 = & \langle\{\mathbf{1} : 0o \leftarrow ro, inO. \\ & inO, & & \lambda : ro \leftarrow not\ fr.\}, \\ & (\neg w, 1)\rangle & & inO, \\ & & & (0o, \lambda)\rangle \end{array}$$

An interesting property of any argument w.r.t. a decision is that the goal that is reached by the argument can be inferred by cutting the possibilistic knowledge based at the certainty level of the goal that is inferred by the given argument. In fact, the certainty of the goal follows the basic principle of *weakest link* [19].

Proposition 3.1. Let $F = \langle P, \mathcal{D}, \mathcal{G}\rangle$ be a possibilistic decision making framework and $A \in \mathcal{A}_F$ such that $A = \langle S, d, (g, \alpha)\rangle$. Then, the following conditions hold:

1. $(g, \alpha) \in T$ such that $P\_WFS(P_\alpha \cup \{1 : d \leftarrow \top.\}) = \langle T, F\rangle$

2. $\alpha = GLB(\{\beta \mid \beta : r \in S\})$

Proof:
Let us start by introducing the following observations:

1. By Theorem 1 and Proposition 4 of [30], we know that if P is a possibilistic definite logic program then $\Pi Cn(P) = \Pi M(P)$. $\Pi M(P)$ denotes the possibilistic model of P such that for all $(x, \alpha) \in \Pi M(P)$, $\alpha = \mathcal{N}_P(a)$ in which $\mathcal{N}_P(a)$ is the necessity measure of a w.r.t. P.

2. Let P be a possibilistic definite logic programs and $\Pi M(P)$ be the possibilistic model of P. Hence by Proposition 11 of [19], $(x, \alpha) \in \Pi M(P)$ if and only if $(x, \alpha) \in \Pi M(P_\alpha)$.

The proof is formalized as following:

1. Let $P' = P \cup \{1 : d \leftarrow \top.\}$. By definition of the possibilistic well founded model, if $(g, \alpha) \in T$ such that $P\_WFS(P') = \langle T, F \rangle$, then $(g, \alpha) \in \Pi Cn(R(P', T^*))$. Therefore, by Observation 1, $\alpha = \mathcal{N}_{R(P',T^*)}(g)$ in which $\mathcal{N}_{R(P,T^*)}(g)$ is the necessity measure of g w.r.t. $R(P, T^*)$. Then, by Observation 2, $(g, \alpha) \in \Pi Cn(R(P', T^*)_\alpha)$. This means that $(g, \alpha) \in T$ such that $P\_WFS(P'_\alpha) = \langle T, F \rangle$

2. The proof directly follows from 1 and Observation 1.

$\square$

Once we have identified the set of arguments of our possibilistic default theory, the relationships between these arguments need to be identified. These relationships are usually captured by the idea of *attack*. In argumentation theory, there are two fundamental notions of attack: *undercut* and *rebut* [35]:

- Undercut is an attack which invalidates an assumption of an argument.
- Rebut is an attack which contradicts a conclusion of an argument.

By keeping in mind that an atom negated by *negation as failure* is an assumption, the attack relation in our framework is defined as follows:

Definition 3.3. Let $A = \langle S_A, d_A, g_A \rangle$, $B = \langle S_B, d_B, g_B \rangle$ be two arguments, $P\_WFS(S_A \cup \{1 : d_A \leftarrow \top.\}) = \langle T_A, F_A \rangle$ and $P\_WFS(S_B \cup \{1 : d_B \leftarrow \top.\}) = \langle T_B, F_B \rangle$. We say that A attacks B if one of the following conditions holds:

- $a \in T_A$ and $\neg a \in T_B$.
- $a \in T_A$ and $a \in F_B$.

Observe that the first condition of the definition is capturing the idea of rebut and the second condition is capturing the idea of undercut. The following example illustrates this definition.

Example 3.3. Let us consider the arguments in Example 3.2. From these arguments, we can see that:[3]

$$
\begin{array}{lll}
P\_WFS(S_1 \cup \{1 : inO \leftarrow \top.\}) = & \langle\{(inO,1),(fr,\gamma),(6O,\gamma)\}, & \{(ro,1)\}\rangle \\
P\_WFS(S_2 \cup \{1 : inC \leftarrow \top.\}) = & \langle\{(inC,1),(fr,\gamma),(6O,\gamma)\}, & \{(ro,1)\}\rangle \\
P\_WFS(S_3 \cup \{1 : inC \leftarrow \top.\}) = & \langle\{(inC,1),(w,1)\}, & \{\}\rangle \\
P\_WFS(S_4 \cup \{1 : tA \leftarrow \top.\}) = & \langle\{(tA,1),(5O,1)\}, & \{\}\rangle \\
P\_WFS(S_5 \cup \{1 : inC \leftarrow \top.\}) = & \langle\{(inC,1),(ro,\lambda),(5O,\lambda)\}, & \{(fr,1)\}\rangle \\
P\_WFS(S_6 \cup \{1 : tA \leftarrow \top.\}) = & \langle\{(tA,1),(\neg w,1)\}, & \{(inC,1)\}\rangle \\
P\_WFS(S_7 \cup \{1 : inO \leftarrow \top.\}) = & \langle\{(inO,1),(\neg w,1)\}, & \{(inC,1)\}\rangle \\
P\_WFS(S_8 \cup \{1 : inO \leftarrow \top.\}) = & \langle\{(inO,1),(ro,\lambda),(0o,\lambda)\}, & \{(fr,1)\}\rangle
\end{array}
$$

From these possibilistic well-founded models, the following attack relations can be identified: $A1$ attacks $A8$, $A1$ attacks $A5$, $A2$ attacks $A5$, $A2$ attacks $A6$, $A2$ attacks $A7$, $A2$ attacks $A8$, $A3$ attacks $A6$, $A3$ attacks $A7$, $A5$ attacks $A6$, $A5$ attacks $A7$, $A6$ attacks $A3$, $A7$ attacks $A3$, $A8$ attacks $A2$ and $A8$ attacks $A1$. A graph representation of these attacks is presented in Figure 1, in which arguments are represented as nodes and attacks as edges.

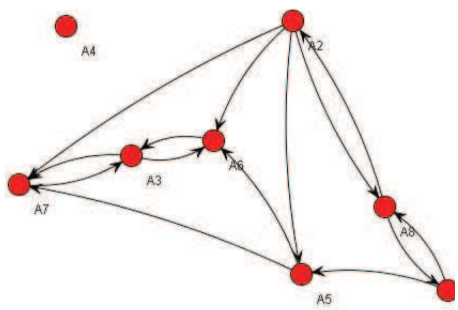

Figure 1. Graph representation.

By having a set of arguments and their relations, a possibilistic decision making framework can be instantiated into a possibilistic argumentation decision making framework. Since any pair of arguments can be compared by different criteria (such as the certainty level of the goal that is reached by the given argument), a possibilistic argumentation decision making framework is provided with a partial order relation. Hence, given a set of arguments $\mathcal{A}_F$, $\preceq_{\mathcal{A}_F}$ denotes a partial order in $\mathcal{A}_F$.

Definition 3.4. A possibilistic argumentation decision making framework is a tuple $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F}\rangle$, where F is a possibilistic decision making framework, and Att denotes the binary relations of attacks (according to Definition 3.3) in $\mathcal{A}_F$, *i.e.* $Att \subseteq \mathcal{A}_F \times \mathcal{A}_F$.

Essentially, a possibilistic argumentation decision making framework is an extension of a possibilistic decision making framework. However, this structure allows us to tackle a decision making problem from

[3]We assume that each argument is of the form $A_i = \langle S_i, d, (g,\alpha)\rangle$.

another perspective. In particular, the problem can be regarded as a set of *pros* and *cons* arguments w.r.t. a set of decisions in which different patterns of selection of arguments can be applied. These patterns of selection are usually called *argumentation semantics*.

In order to define a basic requirement of any argumentation semantics of a possibilistic argumentation decision making framework, the idea of context-free is defined. Given a set of arguments $\mathcal{A}_F$, a subset of arguments $\mathcal{E} \subseteq \mathcal{A}_F$ is context-free if and only if there are no arguments $A, B \in \mathcal{E}$ such that *A attacks B*.

Definition 3.5. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework. A basic argumentation semantics SEM_{Arg} of PF is a function from PF to $2^{2^{\mathcal{A}_F}}$ such that the following condition holds: if $\mathcal{E} \in SEM_{Arg}(PF)$, then $\forall E \in \mathcal{E}$, E is context-free.

Since the property of context-free is considered a fundamental property of any argumentation semantics [8, 35], this is the only restriction that we impose on a basic argumentation semantics. Essentially, a basic argumentation semantics is *a pattern of selection of arguments* such as the argumentation semantics defined in [9, 21, 32].

Observe that SEM_{Arg} is a set of subsets of $\mathcal{A}_F$. Any set of arguments $\mathcal{E}$ which belongs to SEM_{Arg} is called an *extension*. In our setting, an extension denotes a possible solution to a decision making problem.

An interesting property of any extension belonging to a basic argumentation semantics is that the set of goals which are supported by an extension can be inferred by considering a subset of the possibilistic default theory. In order to formalize this property, let us introduce the following notation. Given a set of arguments $\mathcal{A}$:

$$\mathcal{P}(\mathcal{A}) := \{S \cup \{1 : d \leftarrow \top.\} | \langle S, d, (g, \alpha) \rangle \in \mathcal{A}\}$$

$$\mathcal{D}(\mathcal{A}) := \bigsqcup_{\langle S, d, (g,\alpha) \rangle \in \mathcal{A}} \{(g, \alpha)\}^4$$

Proposition 3.2. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework, SEM_{Arg} be a basic argumentation semantics. If $\mathcal{E} \in SEM_{Arg}(PF)$ then $\mathcal{D}(\mathcal{E}) \subseteq T$ such that $P\_WFS(\mathcal{P}(\mathcal{E})) = \langle T, F \rangle$.

Proof:
(Sketch) Let us start by introducing some notation w.r.t. relevant rules. Let P be a possibilistic logic program. P induces a notion of $dependency$ between atoms from $\mathcal{L}_{P^*}$. We say that a *depends immediately on* b if and only if b appears in the body of a clause in P, such that a appears in its head. The two place relation *depends on* is the transitive closure of *depends immediately on*. The set of dependencies of an atom x, denoted by *dependencies-of*(x), corresponds to the set $\{a \mid x \ depends \ on \ a\}$. Given $x \in \mathcal{L}_{P^*}$, $rel\_rul(P, x)$ is the set of relevant rules of P with respect to x, *i.e.* the set of rules that contain an $a \in$ *dependencies-of*(x).

[4]Given two sets of possibilistic atoms A and B: $A \sqcup B = \{(x, \alpha) | (x, \alpha) \in A \ and \ x \notin B^*\} \cup \{(x, \alpha) | x \notin A^* \ and \ (x, \alpha) \in B\} \cup \{(x, \mathcal{LUB}(\{\alpha, \beta\})) | (x, \alpha) \in A$ and $(x, \beta) \in B\}$.

The main part of the proof consists as follows: by Lemma 5.30 of [17], we know that the well founded semantics satisfies relevance. Hence, given a possibilistic normal logic program P and $x \in \mathcal{L}_{P^*}$, $P\_WFS(P)(x) = P\_WFS(rel\_rul(P, x))(x)$. This means that the semantic value of any atom x in $\mathcal{L}_{P^*}$ w.r.t. P_WFS depends on the relevant rules of x. Therefore, since $\mathcal{P}(\mathcal{E})$ are the relevant rules of $\mathcal{D}(\mathcal{E})^*$, the semantic values of $\mathcal{D}(\mathcal{E})^*$ w.r.t. P_WFS depends only on $\mathcal{P}(\mathcal{E})$. □

Observe that $\mathcal{P}(\mathcal{E})$ is denoting the relevant knowledge base of the extension $\mathcal{E}$ and $\mathcal{D}(\mathcal{E})$ denotes a set of decisions which are supported by $\mathcal{P}(\mathcal{E})$. This property of modularity of a given possibilistic default theory w.r.t. a basic argumentation semantics is inherited from the well founded semantics which satisfies the property of *relevance* [17]. It is worth mentioning that relevance is a desired property for performing non-monotonic reasoning [17]. In fact, the property formalized by Proposition 3.2 has practical applications, *e.g.* for the interchange of knowledge between intelligent agents.

In order to characterize a particular argumentation semantics for a possibilistic argumentation framework, we define a version of the preferred semantics (introduced in [21]).

Definition 3.6. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework.

- We say that A defeats B if and only if A attacks B, B does not attack A, and $A \succeq_{\mathcal{A}_F} B$.
- An argument $A \in \mathcal{A}_F$ is acceptable with respect to $S \in \mathcal{A}_F$, if $\forall B \in \mathcal{A}_F$ which defeats A there exists $C \in S$ such that C defeats B.
- A conflict-free $S \subseteq \mathcal{A}_F$ is admissible if and only if each argument in S is acceptable with respect to S.
- A preferred extension of PF is a maximal admissible set of PF. $SEM_{preferred}$ denotes the set of preferred extensions of PF.

We illustrate the inference of the preferred semantics in the following example.

Example 3.4. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework such that F is the possibilistic decision making framework presented in Example 3.1, $\mathcal{A}_F$ is the set of arguments presented in Example 3.2, Att is the set of attacks presented in Example 3.3 and $\preceq_{\mathcal{A}_F}$ is the equality relation.[5] It is easy to see that $SEM_{preferred}(PF) = \{\{A4, A8, A5, A3\}, \{A4, A1, A2, A3\}\}$

Since the preferred semantics' definition is based on admissible sets, the following property is straightforward.

Proposition 3.3. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework. $SEM_{preferred}$ is a basic argumentation semantics of PF.

Proof:
If $\mathcal{E} \in SEM_{preferred}$, then $\mathcal{E}$ is an admissible set; therefore, $\mathcal{E}$ is a conflict-free set. □

In the following subsections, we introduce some criteria for prioritizing/contrasting arguments and extensions.

[5]We assume the equality relation for the sake of simplicity.

4. Preferences between Decisions

So far, we have shown that a decision making problem can be captured in a possibilistic default theory. Such possibilistic default theory can be instantiated in a possibilistic argumentation decision making framework. On the other hand, by applying a basic argumentation semantics to a possibilistic argumentation decision making framework, one can infer different scenarios (called extensions) which represent potential solutions to a given decision making problem. Since an extension has different arguments that argue for a particular decision, some criteria for selecting a suitable decision are worth defining.

4.1. Preferences between Arguments

We start by defining a preference relation between arguments. To this end, we will define the notion of strength of an argument.

Definition 4.1. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework and $A \in \mathcal{A}_F$ such that $A = \langle S, d, (g, \alpha) \rangle$. The strength of A is a pair $\langle Lev(A), Wei(A) \rangle$ such that:

- The certainty level of the argument is $Lev(A) = \alpha$.
- The weight of the argument is $Wei(A) = \beta$ such that $(g, \beta) \in \mathcal{G}$.

The use of a dual value for dealing with the strength of an argument, as it is done by Definition 4.1, was explored in the context of possibilistic theories in [5]. The strength of an argument allows us to compare pairs of arguments. Informally, an argument is all the better as it uses more certain knowledge and refers to an important goal. This can be formally captured by a Pareto-based comparison criterion.

Definition 4.2. Let $PF = \langle F, \mathcal{A}_F, Att, \preceq_{\mathcal{A}_F} \rangle$ be a possibilistic argumentation decision making framework, SEM_{Arg} be a basic argumentation semantics and let $A, B \in \mathcal{E}$ such that $\mathcal{E} \in SEM_{Arg}(PF)$. A is stronger than B (denoted $A \succ_p B$) if and only if $\langle Lev(A), Wei(A) \rangle >_{pareto} \langle Lev(B), Wei(B) \rangle$

The condition in the above definition follows the principle of Pareto optimality according to which an argument is preferred if it is better or equal to another in all attributes and strictly better in at least one attribute. The set of best arguments is represented by the Pareto frontier which contains arguments which are not dominated by any other arguments. A way for computing the Pareto frontier is by means of the *skyline operator* [12].

It is important to observe that the Pareto relation can be used for defining the acceptability of arguments. This means that a possibilistic argumentation decision making framework can be instantiated as $PF = \langle F, \mathcal{A}_F, Att, \prec_p \rangle$. In this case, any basic argumentation semantics applying to PF could use $>_{pareto}$ for defining the acceptability of arguments from $\mathcal{A}_F$ (see the concept of defeat presented in Definition 3.6).

We illustrate the idea of Pareto relation in the following example.

Example 4.1. Let $PF = \langle F, \mathcal{A}_F, Att, \prec_p \rangle$ be a possibilistic argumentation decision making framework such that F is the possibilistic decision making framework presented in Example 3.1, $\mathcal{A}_F$ is the set of arguments presented in Example 3.2, Att is the set of attacks presented in Example 3.3, and $\prec_p$ is the preference/acceptability relation of arguments according to Definition 4.2. Let us consider preferred extensions $E_1 = \{A4, A8, A5, A3\}$ and $E_2 = \{A4, A1, A2, A3\}$ in Example 3.4. Then:

- in E_1, $A_3 \succ_p A_4 \succ_p A_5 \succ_p A_8$
- in E_2, $A_3 \succ_p \{A_4, A_1, A_2\}$

Therefore, the best argument for both extensions is A_3.

More generally, the following property holds.

Proposition 4.1. Let $PF = \langle F, \mathcal{A}_F, Att, \prec_p \rangle$ be a possibilistic argumentation decision making framework, SEM_{Arg} be a basic argumentation semantics and $\mathcal{E} \in SEM_{Arg}(PF)$. Then, $\prec_p$ is a partial order relation over arguments in $\mathcal{E}$.

Proof:
Straightforward by Pareto ordering definition. □

4.2. Preferences between Extensions

Sometimes, it is desirable to compare extensions based on the arguments which support a decision d with a level of certainty α or a priority level β of satisfied goals. There exist many different ways to induce a preference relation over extensions.

To this end, we define for a decision d the set of arguments in an extension E as $E^i_X(d) = \{A \in E \mid A = \langle S, d, (g, \alpha) \rangle \wedge X(A) = i\}$, in which $X(A)$ can be either the level or the weight of an argument according to Definition 4.1.

The first comparison criterion is based on the cardinality of the set of arguments which satisfy a goal to a particular degree.

Definition 4.3. Let E_1 and E_2 be two extensions of a possibilistic argumentation decision making framework PF. Given a decision d, $E_1(d)$ is cardinality-preferred to $E_2(d)$ ($E_1(d) \succ_c E_2(d)$) if and only if there exists a maximal i such that $|{E^i_1}_X(d)| \neq |{E^i_2}_X(d)|$ and $|{E^i_1}_X(d)| > |{E^i_2}_X(d)|$.

In certain applications, counting does not provide the best way of defining an order between extensions. Therefore, a more cautious preference relation can be defined (in the sense that fewer extensions are considered better than others) based on set inclusion of the arguments that support a decision.

Definition 4.4. Let E_1 and E_2 be two extensions of a possibilistic argumentation decision making framework PF. Given a decision d, $E_1(d)$ is inclusion-preferred to $E_2(d)$ ($E_1(d) \succ_i E_2(d)$) if and only if a maximal i exists such that ${E^i_1}_X(d) \neq {E^i_2}_X(d)$ and ${E^i_1}_X(d) \supset {E^i_2}_X(d)$.

More generally, the following property holds.

Proposition 4.2. Let E_1 and E_2 be two extensions of a possibilistic argumentation decision making framework PF. Given a decision d, then $E_1(d) \succ_i E_2(d)$ implies $E_1(d) \succ_c E_2(d)$.

Proof:
Ab absurdo: let us suppose that $E_1(d) \succ_i E_2(d)$ does not imply $E_1(d) \succ_c E_2(d)$. Thus, given a maximal i and $A(X) = Lev(A)$ (resp. $A(X) = Wei(A)$) there must exist an element $a \in E^i_{2_{Lev}}(d)$ such that either $|E^i_{2_{Lev}}(d)| = |E^i_{1_{Lev}}(d)|$ or $|E^i_{2_{Lev}}(d)| > |E^i_{1_{Lev}}(d)|$. But both theses contradict the hypothesis that the set $E^i_{1_{Lev}}(d)$ contains more elements than $E^i_{2_{Lev}}(d)$. □

5. Related work

In the literature, argumentation has been used for different purposes and a good survey is presented in [10]. Among the argumentation proposals, several frameworks address inconsistency in knowledge bases [4], non-monotonic reasoning [21], and qualitative decision making under uncertainty [5]. Since the argumentation framework proposed in this paper aims to address decision making under uncertainty, the closest works to our proposal are, to the best of our knowledge, the proposals of Amgoud and Prade in [5] and Alsinet *et al.*in [2].

Amgoud and Prade in [5] have proposed a unified argumentation-based model for different decision problems such as decision making under uncertainty, multiple criteria decisions, and rule-based decisions. The proposed framework follows two main steps: the building of arguments and their comparison by means of decision criteria. These criteria support the definition of a complete preorder on a set of candidate decisions based on the inferred arguments. One of the nicest features of their approach is the definition of a complete and unified framework. However, it seems to lack a practical point of view (at least in an efficient way). The inference of arguments is indeed based on the inference of a purely logical approach, whereas in our approach arguments inference is done on the basis of the non-monotonic inference of the possibilistic well founded semantics (which is polynomial computable). Another remarkable difference is in the argument selection supporting the best decision. In fact, in their framework, a pessimistic and optimistic criteria, from a qualitative decision making under uncertainty point of view [20], are used to select the best argument. Instead, in our approach, argument selection is based on the strength of argument only.

On the other hand, the proposal of Alsinet *et al.*, in [2], is an argumentation approach which combines features from argumentation theory and logic programming (without negation as failure). The specification language of this approach is based on a possibilistic logic programming approach. It is worth mentioning that this possibilistic logic programming approach is defined over the possibilistic $Gödel$ logic [3]. Indeed, the construction of arguments is based on the inference of the possibilistic $Gödel$ logic. The inference over conflicting arguments is based on a dialectical analysis which defines a skeptical reasoning process.

In [15], a probabilistic argumentation approach was proposed. This approach is useful when the application domain permits the definition of *probabilistic* links between *premises* and *conclusions* of an argument. In our definition of argument, we also make a direct relation between the degree of certainty of the argument's conclusion (the reached goal) and the degree of certainty of the argument's premises (see Proposition 3.1). It is important to point out that sometimes, when a probability approach is used, one of the hardest parts of solving a problem is identifying the probability relations[6]. However, sometimes it is enough to have just relative likelihoods for modeling different levels of evidence/uncertainty *e.g. possible*, *probable*, *plausible*, *etc.*, in which each relative likelihood is a possible class of beliefs. In

[6]We refer to [28] for a discussion about the difficulties related of finding a numerical representation for uncertainty.

this case, the possibilistic approaches such as the suggested approach in this paper and in [2, 5] take relevance.

6. Conclusions

In this paper, we have proposed an argumentation-based possibilistic decision making framework which is able to capture uncertain information and exceptions/defaults. In fact, one of the noticeable features in our approach, as we have claimed in the paper, is that we can deal with a reasoning that is at the same time *non-monotonic* and *uncertain*. Unlike standard argumentation approaches which are based on the monotonic inference of classical logic for building arguments, our approach is based on the non-monotonic inferences of the possibilistic well founded model. As claimed before, this semantics owns some nice features such as its complexity class which is polynomial. As a result, since the construction of arguments is based on the possibilistic Well-Founded semantics, we have obtained a computable argumentation framework for building arguments, constructing attack relations, and comparing set of arguments, and extensions.

We have defined the concept of a possibilistic decision making framework, which is based on a possibilistic default theory, a set of decisions and a prioritized set of goals. This set of goals captures user preferences in achieving a particular state in a decision making problem.

We have defined the concept of argument w.r.t. a decision by considering the inference of the possibilistic well-founded semantics. This argument captures the feasibility of reaching a goal by applying a decision in a given context. An interesting property of any argument w.r.t. a decision is that the goal that is reached by the argument can by inferred by cutting the possibilistic knowledge based at certainty level of the goal that is inferred by the given argument. In fact, the certainty of the goal follows the basic principle of *weakest link* (Proposition 3.1).

Since the possibilistic well-founded semantics is a three valued semantics, the relation of attacks between arguments has been defined in terms of complementary atoms and assumptions. This relation of attack captures the idea of *rebut* and *undercut*.

The inference in our argumentation-based decision making framework relies on basic argumentation semantics (following Dung's style). An interesting property of any extension which belongs to a basic argumentation semantics is that the set of goals which are supported by an extension can be inferred by considering a subset of the possibilistic default theory (Proposition 3.2). This property of modularity w.r.t. a given possibilistic default theory of any basic argumentation semantics is inherited from the well founded semantics which satisfies the property of *relevance*.

We have been working in the support of decision making process for deciding whether an industrial wastewater is safe for being discharged into the system [6, 25]. Our experience suggests that in the environmental domain, we require a *qualitative* theory of *default reasoning* for modeling incomplete information and an uncertain theory like possibilistic logic for modeling uncertain events. Hence, as part of our future work, we will consider applying our possibilistic decision making framework to the domain of environmental systems.

Acknowledgements

We are grateful to anonymous referees for their useful comments.

References

[1] Alsinet, T., Chesñevar, C. I., Godo, L., Sadri, S., Simari, G. R.: Formalizing Argumentative Reasoning in a Possibilistic Logic Programming Setting with Fuzzy Unification, *International Journal of Approximate Reasoning*, **48**(3), August 2008, 711–729.

[2] Alsinet, T., Chesñevar, C. I., Godo, L., Simari, G. R.: A logic programming framework for possibilistic argumentation: Formalization and logical properties, *Fuzzy Sets and Systems*, **159**(10), 2008, 1208–1228.

[3] Alsinet, T., Godo, L.: A Complete Calculus for Possibilistic Logic Programming with Fuzzy Propositional Variables, *Proceedings of the Sixteen Conference on Uncertainty in Artificial Intelligence*, ACM Press, 1-10, 2000.

[4] Amgoud, L., Cayrol, C.: Inferring from Inconsistency in Preference-Based Argumentation Frameworks, *Journal of Automated Reasoning*, **29**(2), 2002, 125–169.

[5] Amgoud, L., Prade, H.: Using arguments for making and explaining decisions, *Artificial Intelligence*, **173**(3-4), March 2009, 413–436.

[6] Aulinas, M., Nieves, J. C., Poch, M., Cortés, U.: Supporting Decision Making in Urban Wastewater Systems using a Knowledge-Based Approach, *Environmental Modelling and Software*, **26**(5), 2011, 562–572.

[7] Baral, C.: *Knowledge Representation, Reasoning and Declarative Problem Solving*, Cambridge University Press, Cambridge, 2003.

[8] Baroni, P., Giacomin, M.: On principle-based evaluation of extension-based argumentation semantics, *Artificial Intelligence.*, **171**(10-15), 2007, 675–700.

[9] Baroni, P., Giacomin, M., Guida, G.: SCC-recursiveness: a general schema for argumentation semantics, *Artificial Intelligence*, **168**(1-2), October 2005, 162–210.

[10] Bench-Capon, T. J. M., Dunne, P. E.: Argumentation in artificial intelligence, *Artificial Intelligence*, **171**(10-15), 2007, 619–641.

[11] Bonet, B., Geffner, H.: Arguing for Decisions: A Qualitative Model of Decision Making, *Proceedings of the 12th Annual Conference on Uncertainty in Artificial Intelligence (UAI-96)*, Morgan Kaufmann Publishers, San Francisco, CA, 1996, 98–105.

[12] Börzsönyi, S., Kossmann, D., Stocker, K.: The Skyline Operator, *Proceedings of the 17th International Conference on Data Engineering*, IEEE Computer Society, Washington, DC, USA, 2001, ISBN 0-7695-1001-9, 421–430.

[13] Brass, S., Zukowski, U., Freitag, B.: Transformation-Based Bottom-Up Computation of the Well-Founded Model., *NMELP*, 1996, 171–201.

[14] Brewka, G.: Answer Sets and Qualitative Decision Making, *Synthese*, **146**, 2005, 171–187, ISSN 0169-2968.

[15] Carofiglio, V.: Modelling Argumentation with Belief Networks, *CMNA IV, 4^{th} Workshop on Computational Models of Natural Argument* (F. Grasso, C. Reed, G. Carenini, Eds.), ECAI 2004, Valencia, Spain, 2004.

[16] Dix, J.: A Classification Theory of Semantics of Normal Logic Programs: I. Strong Properties., *Fundamenta Informaticae*, **22**(3), 1995, 227–255.

[17] Dix, J.: A Classification Theory of Semantics of Normal Logic Programs: II. Weak Properties., *Fundamenta Informaticae*, **22**(3), 1995, 257–288.

[18] Dix, J., Osorio, M., Zepeda, C.: A general theory of confluent rewriting systems for logic programming and its applications., *Annals of Pure and Applied Logic*, **108**(1-3), 2001, 153–188.

[19] Dubois, D., Lang, J., Prade, H.: Possibilistic Logic, in: *Handbook of Logic in Artificial Intelligence and Logic Programming, Volume 3: Nonmonotonic Reasoning and Uncertain Reasoning* (D. Gabbay, C. J. Hogger, J. A. Robinson, Eds.), Oxford University Press, Oxford, 1994, 439–513.

[20] Dubois, D., Prade, H.: Possibility theory as a basis for qualitative decision theory, *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995, ISBN 1-55860-363-8, 1924–1930.

[21] Dung, P. M.: On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games., *Artificial Intelligence*, **77**(2), 1995, 321–358.

[22] Dung, P. M., Thang, P. M., Toni, F.: Towards argumentation-based contract negotiation, *Second International Conference on Computational Models of Argument (COMMA'08)*, 172, IOS Press, 2008, 134–146.

[23] Fox, J., Glasspool, D., Grecu, D., Modgil, S., South, M., Patkar, V.: Argumentation-Based Inference and Decision Making — A Medical Perspective, *IEEE Intelligence Systems*, **22**(6), November/December 2007, 34–41.

[24] Fox, J., Parsons, S.: *Applications of Uncertainty Formalisms*, chapter Arguing about beliefs and actions, Springer-Verlag, Berlin, 1998, 266–302.

[25] Garcia, D., Nieves, J. C., Cortés, U.: Reasoning about Actions for the Management of Urban Wastewater Systems using a Causal Logic, *International Congress on Environmental Modelling and Software*, Ottawa, Canada, 2010.

[26] Gelder, A. V., Ross, K. A., Schlipf, J. S.: The Well-Founded Semantics for General Logic Programs., *Journal of the ACM*, **38**(3), 1991, 620–650.

[27] Gelfond, M., Lifschitz, V.: The Stable Model Semantics for Logic Programming, *5th Conference on Logic Programming* (R. Kowalski, K. Bowen, Eds.), MIT Press, 1988, 1070–1080.

[28] Halpern, J. Y.: *Reasoning about uncertainty*, The MIT Press, 2005.

[29] Krause, P., Ambler, S., Elvang-Gøransson, M., Fox, J.: A Logic of Argumentation for Reasoning under Uncertainty., *Computational Intelligence*, **11**, 1995, 113–131.

[30] Nicolas, P., Garcia, L., Stéphan, I., Lefèvre, C.: Possibilistic Uncertainty Handling for Answer Set Programming, *Annals of Mathematics and Artificial Intelligence*, **47**(1-2), June 2006, 139–181.

[31] Nieves, J. C., Osorio, M., Cortés, U.: Semantics for Possibilsitic Disjuntive Programs, *Theory and Practice of Logic Programming*, 2011, DOI: 10.1017/S1471068411000408.

[32] Nieves, J. C., Osorio, M., Zepeda, C.: A Schema for Generating Relevant Logic Programming Semantics and its Applications in Argumentation Theory, *Fundamenta Informaticae*, **106**(2-4), 2011, 295–319.

[33] Osorio, M., Nieves, J. C.: Possibilistic Well-Founded Semantics, *8th Mexican International Conference on Artificial Intelligence, MICAI'09*, 5845, Springer-Verlag, 2009, 15–26.

[34] Pelletier, F. J., Elio, R.: *Scope of Logic, Methodology and Philosophy of Science*, vol. 1 of *Synthese Library*, chapter Logic and Computation, Dordrecht: Kluwer Academic Press, 2002, 137–156.

[35] Prakken, H., Vreeswijk, G. A. W.: Logics for defeasible argumentation, in: *Handbook of Philosophical Logic* (D. Gabbay, F. Günthner, Eds.), vol. 4, second edition, Kluwer Academic Publishers, Dordrecht/Boston/London, 2002, 219–318.

[36] Savage, L. J.: *The Foundations of Statistics*, Dover Publications, Inc., New York, 1972.

[37] Toni, F.: Argumentative Agents, *International Multiconference on Computer Science and Information Technology*, 5, 2010.

[38] Wooldridge, M.: *Reasoning about Rational Agents*, MIT Press, Cambridge, USA, 2000.