Group Decis Negot (2014) 23:979–1012
DOI 10.1007/s10726-012-9324-4

Arguing over Goals for Negotiation: Adopting an Assumption-Based Argumentation Decision Support System

Maxime Morge · **Paolo Mancarella**

Published online: 15 December 2012

Abstract Computational models of argumentation has been put forward as a promising approach to support decision making. In this context several recent works have proposed argumentation-based frameworks for decision making. In this paper we describe an application based on an argumentation-based mechanism for decision-making to concede. Adopting the assumption-based approach of argumentation, we propose an argumentation framework in which preferences are attached to goals. Arguments are defined as tree-like structures. Our framework is equipped with a computational counterpart for solving a decision problem, modeling the intuition that high-ranked goals are preferred to low-ranked goals which can be withdrawn. In this way, our framework suggests some decisions and provides an interactive and intelligible explanation of this choice. Our implementation, called MARGO, has been used for service selection within the ArguGRID project. We illustrate our approach with an industrial application, and illustrate the operation of the system with a running example.

Keywords Negotiation · Artificial intelligence · Multi-agent systems · Computational model of argumentation

M. Morge (✉)
Laboratoire d'Informatique Fondamentale de Lille, Université Lille 1, 59650 Villeneuve d'Ascq, France
e-mail: Maxime.Morge@univ-lille1.fr

P. Mancarella
Dipartimento di Informatica, Largo B. Pontecorvo, 3, 56127 Pisa, Italy
e-mail: Paolo.Mancarella@unipi.it

1 Introduction

Negotiations occur in procurement, commerce, health and government, among organisations (companies and institutions) and individuals. For instance, electronic procurement (respectively electronic commerce) consists of business-to-business (respectively business-to-customer) purchase and provision of resources or services through the Internet. Typically, organisations and individuals invite bids and negotiate costs, volume discounts or special offers. These negotiations can be (at least partially) delegated to software components in order to reach agreements (semi-)automatically (Jennings et al. 2001). For this purpose, software agents must be associated with stakeholders in negotiations.

In negotiations, participation is voluntary and there is no third party imposing a resolution of conflicts. Participants resolve their conflict by verbal means. The aim for all parties is to "make a deal" while bargaining over their interests, typically seeking to maximise their "good" (welfare), and prepared to concede some aspects while insisting on others. Each side tries to figure out what other sides may want most, or may feel is most important. Since real-world negotiations can be resolved by confronting and evaluating the justifications of different positions, argumentation can support such a process. Logical models of argument (Chesñevar et al. 2000) can be used to support rational decision making by agents, to guide and empower negotiation amongst stakeholders and allow them to reach agreements. With the support of argumentation processes, agents decide which agreements can be acceptable to fulfil the requirements of users and the constraints imposed by interlocutors, taking into account their expertises/preferences and the utilities they assign to situations. This is the reason why many works in the area of Artificial Intelligence focus on computational models of argumentation-based negotiation (Rahwan et al. 2003). Logical models of arguments (e.g. Kakas and Moraitis 2003; Bench-Capon and Prakken 2006; Amgoud and Prade 2009) can be used to encompass the reasoning of agents engaged in negotiations. However, these approaches do not come with a mechanism allowing interacting agents to concede. Since agents can consider multiple goals which may not be fulfilled all together by a set of non-conflicting decisions, e.g. a negotiation agreement, high-ranked goals must be preferred to low-ranked goals on which agents can concede. In this paper we propose an argumentation-based mechanism for decision-making to concede. Adopting the assumption-based approach of argumentation, we propose here an argumentation framework. It is built upon a logic language which holds statements representing knowledge, goals, and decisions. Preferences are attached to goals. These concrete data structures consist of information providing the backbone of arguments. Due to the abductive nature of practical commonsense reasoning, arguments are built by reasoning backwards. Moreover, arguments are defined as tree-like structures. Our framework is equipped with a computational counterpart (in the form of a formal mapping from it into a set of assumption-based argumentation frameworks). Indeed, we provide the mechanism for solving a decision problem, modeling the intuition that high-ranked goals are preferred to low-ranked goals which can be withdrawn. Thus, we give a clear semantics to the decisions. In this way, our framework suggests some decisions and provides an interactive and intelligible explanation of this choice. Our implementation, called MARGO, is a tool for multi-attribute qualitative decision-making as required, for instance in agent-based negotiation or in service-oriented agents. In a more practical context, our framework is amenable to industrial applications. In particular, MARGO has been used within the ArguGRID project[1] for service selection and service negotiation.

The paper is organised as follows. Section 2 introduces the basic notions of argumentation in the background of our work. Section 3 defines the core of our proposal, i.e. our argumentation-based framework for decision-making. Firstly, we define the framework which captures decision problems. Secondly, we define the arguments. Thirdly, we formalize the interactions amongst arguments in order to define our AF (Argumentation Framework). Finally, we provide the computational counterpart of our framework. Section 4 outlines the implementation of our AF and its usage for service-oriented negotiation (cf Sect. 5). Finally, Sect. 6 discusses some related works and Sect. 7 concludes with some directions for future work.

2 Background

2.1 Abstract Argumentation

Our argumentation approach is based on Dung's abstract approach to defeasible argumentation (Dung 1995). Argumentation is abstractly defined as the interaction amongst arguments, reasons supporting claims, which can be disputed by other arguments. In his seminal work, Dung considers arguments as atomic and abstract entities interacting through a binary relation over these interpreted as "the argument x attacks the argument y". More formally, an abstract argumentation framework (AAF for short) is defined as follows.

Definition 1 (*AAF*) An *abstract argumentation framework* is a pair $\mathtt{aaf} = \langle \mathcal{A}, \mathtt{attacks} \rangle$ where $\mathcal{A}$ is a finite set of arguments and $\mathtt{attacks} \subseteq \mathcal{A} \times \mathcal{A}$ is a binary relation over $\mathcal{A}$. When $(\mathtt{a}, \mathtt{b}) \in \mathtt{attacks}$, we say that $\mathtt{a}$ attacks $\mathtt{b}$. Similarly, we say that the set $\mathtt{S}$ of arguments attacks $\mathtt{b}$ when $\mathtt{a} \in \mathtt{S}$.

This framework is abstract since it specifies neither the nature of arguments nor the semantics of the attack relation. However, an argument can be viewed as a reason supporting a claim which can be challenged by other reasons.

According to this framework, Dung introduces various extension-based semantics in order to analyse when a set of arguments can be considered as collectively justified.

Definition 2 (*Semantics*) Let $\mathtt{aaf} = \langle \mathcal{A}, \mathtt{attacks} \rangle$ be an abstract argumentation framework. For $\mathtt{S} \subseteq \mathcal{A}$ a set of arguments, we say that:

- $\mathtt{S}$ is *conflict-free* iff $\forall \mathtt{a}, \mathtt{b} \in \mathtt{S}$ it is not the case that $\mathtt{a}$ attacks $\mathtt{b}$;
- $\mathtt{S}$ is *admissible* (denoted $\mathtt{adm}_{\mathtt{aaf}}(\mathtt{S})$) iff $\mathtt{S}$ is conflict-free and $\mathtt{S}$ attacks every argument $\mathtt{a}$ such that $\mathtt{a}$ attacks some arguments in $\mathtt{S}$;
- $\mathtt{S}$ is *preferred* iff $\mathtt{S}$ is maximally admissible (with respect to set inclusion);
- $\mathtt{S}$ is *complete* iff $\mathtt{S}$ is admissible and $\mathtt{S}$ contains all arguments $\mathtt{a}$ such that $\mathtt{S}$ attacks all attacks against $\mathtt{a}$;

[1] http://www.argugrid.eu.

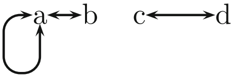
- $\mathtt{S}$ is *grounded* iff $\mathtt{S}$ is minimally complete (with respect to set inclusion);
- $\mathtt{S}$ is *ideal* iff $\mathtt{S}$ is admissible and it is contained in every preferred set.

These declarative model-theoretic *semantics* of the AAF capture various degrees of justification, ranging from very permissive conditions, called *credulous*, to restrictive requirements, called *sceptical*. The semantics of an admissible (or preferred) set of arguments is credulous, in that it sanctions a set of arguments as acceptable if it can successfully dispute every arguments against it, without disputing itself. However, there might be several conflicting admissible sets. That is the reason why various sceptical semantics have been proposed for the AAF, notably the grounded semantics and the sceptically preferred semantics, whereby an argument is accepted if it is a member of all maximally admissible (preferred) sets of arguments. The ideal semantics was not present in (Dung 1995), but it has been proposed recently (Dung et al. 2007) as a less sceptical alternative than the grounded semantics but it is, in general, more sceptical than the sceptically preferred semantics.

Example 1 (AAF) In order to illustrate the previous notions, let us consider the abstract argumentation framework $\mathtt{aaf} = \langle \mathcal{A}, \ \mathtt{attacks} \ \rangle$ where:

- $\mathcal{A} = \{a, b, c, d\}$;
- $\mathtt{attacks} \ = \{(a, a), (a, b), (b, a), (c, d), (d, c)\}$.

The following graph represents this AAF, whereby the fact that "x $\mathtt{attacks}$ y" is depicted by a directed arrow from x to y:



We can notice that:

- {} is grounded;
- $\{b, d\}$ and $\{b, c\}$ are preferred.
- $\{b\}$ is the maximal ideal set.

As previously mentioned, Dung's seminal calculus of opposition deals neither with the nature of arguments nor with the semantics of the attacks relation.

Unlike the abstract argumentation, assumption-based argumentation considers neither the arguments nor the attack relations as primitives. Arguments are built by reasoning backwards from conclusions to assumptions given a set of inference rules. Moreover, the attack relation is defined in terms of a notion of "contrary"(Bondarenko et al. 1993; Dung et al. 2007). Actually, assumption-based argumentation frameworks (ABFs, for short) are concrete instances of AAFs built upon deductive systems.

The abstract view of argumentation does not deal with the problem of finding arguments and attacks amongst them. Typically, arguments are built by joining rules, and attacks arise from conflicts amongst such arguments.

Definition 3 (*DS*) A deductive system is a pair $(\mathcal{L}, \mathcal{R})$ where

- $\mathcal{L}$ is a formal language consisting of countably many sentences, and
- $\mathcal{R}$ is a countable set of inference rules of the form r: $\alpha \ \leftarrow \ \alpha_1, \ldots, \alpha_n$ where $\alpha \in \mathcal{L}$, called the *head* of the rule (denoted $\mathtt{head}(r)$), $\alpha_1, \ldots, \alpha_n \in \mathcal{L}$, called the *body* (denoted $\mathtt{body}(r)$), and $n \geq 0$.

If $n = 0$, then the inference rule represents an axiom (written simply as α). A deductive system does not distinguish between domain-independent axioms/rules, which belong to the specification of the logic, and domain-dependent axioms/rules, which represents a background theory.

Due to the abductive nature of the practical reasoning, we define and construct arguments by reasoning backwards. Therefore, arguments do not include irrelevant information such as sentences not used to derive a conclusion.

Definition 4 (*Deduction*) Given a deductive system $(\mathcal{L}, \mathcal{R})$ and a selection function f, a (backward) deduction of a conclusion α based on a set of premises P is a sequence of sets $S_1, \ldots, S_m$, where $S_1 = \{\alpha\}$, $S_m = \{P\}$, and for every $1 \leq i < m$, where σ is the sentence occurring in S_i selected by f:

1. if σ is not in P then $S_{i+1} = S_i - \{\sigma\} \cup S$ for some inference rule of the form $\sigma \leftarrow S$ in the set of inference rules $\mathcal{R}$;
2. if σ is in P then $S_{i+1} = S_i$.

Deductions are the basis for the construction of arguments in assumption-based argumentation. In order to obtain an argument from a backward deduction, we restrict the premises to those ones that are taken for granted (called *assumptions*). Moreover, we need to specify when one sentence *contraries* an assumptions to specify when one argument attacks another. In this respect, an ABF considers a deductive system augmented by a non-empty set of assumptions and a (total) mapping from assumptions to their contraries. In order to perform decision making, we consider the generalisation of the original assumption-based argumentation framework and the computational mechanisms, whereby multiple contraries are allowed (Gartner and Toni 2007).

Definition 5 (*ABF*) An *assumption-based argumentation framework* is a tuple $\texttt{abf} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}sm, \mathcal{C}on \rangle$ where:

- $(\mathcal{L}, \mathcal{R})$ is a deductive system;
- $\mathcal{A}sm \subseteq \mathcal{L}$ is a non-empty set of *assumptions*. If $x \in \mathcal{A}sm$, then there is no inference rule in $\mathcal{R}$ such that x is the head of this rule;
- $\mathcal{C}on\colon \mathcal{A}sm \rightarrow 2^{\mathcal{L}}$ is a (total) mapping from assumptions into set of sentences in $\mathcal{L}$, i.e. their *contraries*.

In the remainder of the paper, we restrict ourselves to finite deduction systems, i.e. with finite languages and finite set of rules. For simplicity, we restrict ourselves to flat frameworks (Bondarenko et al. 1993), i.e. whose assumptions do not occur as conclusions of inference rules, such as logic programming or the argumentation framework proposed in this paper.

In the assumption-based approach, the set of assumptions supporting a conclusion encapsulates the essence of the argument.

Definition 6 (*Argument*) An *argument* for a conclusion is a deduction of that conclusion whose premises are all assumptions. We denote an argument $\texttt{a}$ for a conclusion α supported by a set of assumptions $\texttt{A}$ simply as $\texttt{a}\colon \texttt{A} \vdash \alpha$.

The set of arguments built upon $\mathcal{A}sm$ is denoted $\mathcal{A}(\mathcal{A}sm)$.

In an assumption-based argumentation framework, the attack relation amongst arguments comes from the contrary relation.

Definition 7 (*Attack relation*) An argument $\texttt{a}\colon \texttt{A} \vdash \alpha$ *attacks* an argument $\texttt{b}\colon \texttt{B} \vdash \beta$ iff there is an assumption $x \in \texttt{B}$ such as $\alpha \in \mathcal{C}on(x)$.

According to the two previous definitions, an ABF is clearly a concrete instantiation of an AAF where arguments are deductions and the attack relation comes from the contrary relation.

Example 2 (ABF) Let $\texttt{abf} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}sm, \mathcal{C}on \rangle$ be an assumption-based argumentation framework where:

- $(\mathcal{L}, \mathcal{R})$ is a deductive system where,
  - $\mathcal{L} = \{\alpha, \beta, \delta, \gamma, \neg\alpha, \neg\beta, \neg\delta, \neg\gamma\}$,
  - $\mathcal{R}$ is the following set of rules,

$$\begin{aligned} \neg\alpha &\leftarrow \alpha \\ \neg\alpha &\leftarrow \beta \\ \neg\beta &\leftarrow \alpha \\ \neg\gamma &\leftarrow \delta \\ \neg\delta &\leftarrow \gamma \end{aligned}$$

- $\mathcal{A}sm = \{\alpha, \beta, \gamma, \delta\}$. Notice that no assumption is the head of an inference rule in $\mathcal{R}$;
- and $\mathcal{C}on(\alpha) = \{\neg\alpha\}$, $\mathcal{C}on(\beta) = \{\neg\beta\}$, $\mathcal{C}on(\gamma) = \{\neg\gamma\}$, and $\mathcal{C}on(\delta) = \{\neg\delta\}$.

Some of the arguments in abf are the following:

$$\begin{aligned} &\{\alpha\} \vdash \neg\alpha \\ &\{\alpha\} \vdash \neg\beta \\ &\{\beta\} \vdash \neg\alpha \\ &\{\gamma\} \vdash \neg\delta \\ &\{\delta\} \vdash \neg\gamma \end{aligned}$$

As stated in Dung et al. (2007), this ABF is a concrete instance of the AAF example proposed previously.

3 Proposal

This section presents our framework to perform decision making. Taking into account its goals and preferences, an agent needs to solve a decision-making problem where the decision amounts to an alternative it can select even if some goals cannot be reached. This agent uses argumentation in order to assess the suitability of alternatives and to identify “optimal” ones. It argues internally to link the alternatives, their features and the benefits that these features guarantee under possibly incomplete knowledge.

We present here the core of our proposal, i.e. an argumentation framework for decision making. Section 3.1 introduces the walk-through example. Section 3.2 introduces

Table 1 The four concrete services and their features

Service	Price	Resolution	DeliveryTime
s(a)	high	low	high
s(b)	high	high	high
s(c)	high	low	low
s(d)	low	low	low

the framework used to capture decision problems. Section 3.3 defines the arguments. Section 3.4 defines the interactions amongst our arguments. Section 3.5 defines our AF. Finally, Sect. 3.6 presents its computational counterpart.

3.1 Walk-Through Example

We consider e-procurement scenarios where buyers seek to purchase earth observation services from sellers (Stournaras 2007). Each agent represents a user, i.e. a service requester or a service provider. The negotiation of the fittest image is a complex task due to the number of possible choices, their characteristics and the preferences of the users. Therefore, this usecase is interesting enough for the evaluation of our argumentation-based mechanism for decision-making (Bromuri et al. 2009; Morge and Mancarella 2010). For simplicity, we abstract away from the real world data of these features and we present here an intuitive and illustrative scenario.

In our scenario, we consider a buyer that seeks to purchase a service s(x) from a seller. The latter is responsible for the four following concrete instances of services: s(a), s(b), s(c) and s(d). These four concrete services reflect the combinations of their features (cf Table 1). For instance, the price of s(a) is high (Price(a, high)), its resolution is low (Resolution(a, low)) and its delivery time is high (DeliveryTime(a, high)). According to the preferences and the constraints of the user represented by the buyer: the cost must be low (cheap); the resolution of the service must be high (good); and the delivery time must be low (fast). Additionally, the buyer is not empowered to concede about the delivery time but it can concede indifferently about the resolution and/or the cost. According to the preferences and constraints of the user represented by the seller: the cost of the service must be high; the resolution of the service must be low; and the delivery time must be high (slow). The seller is not empowered to concede about the cost but it can concede indifferently about the resolution or/and the delivery time. The agents attempt to come to an agreement on the contract for the provision of a service s(x). Taking into account some goals, preferences and constraints, the buyer (resp. the seller) needs to interactively solve a decision-making problem where the decision amounts to a service it can buy (resp. provide).

The decision problem of the buyer can be captured by an abstract argumentation framework which contains the following arguments:

- d_1—He will buy s(d) if the seller accepts it since the cost is low;
- d_2—He will buy s(d) if the seller accepts it since the delivery time is low;
- c—He will buy s(c) if the seller accepts it since the delivery time is low;

Due to the mutual exclusion between the alternatives, c attacks d_1, c attacks d_2, d_1 attacks c and d_2 attacks c. We will illustrate our concrete argumentation framework for decision making with the decision problem of the buyer.

3.2 Decision Framework

Since we want to provide a computational model of argumentation for decision making and we want to instantiate it for particular problems, we need to specify a particular language, allowing us to express statements about the various different entities involved in the knowledge representation for decision making. In our framework, the knowledge is represented by a logical theory built upon an underlying logic-based language.

In this language we distinguish between several different categories of predicate symbols. First of all, we use *goals* to represent the possible objectives of the decision making process. For instance, the goal `fast` represents the objective of a buyer who would like to obtain a quick answer. We will denote by $\mathcal{G}$ the set of predicate symbols denoting goals.

In the language we also want to distinguish symbols representing the *decisions* an agent can adopt. For instance, in the procurement example a unary predicate symbol $\mathtt{s}(x)$ can be used to represent the decision of the buyer to select the service x. It is clear that a problem may involve some decisions over different items, which will correspond to adopting many decision predicate symbols (this is not the case in our running example). We will denote by $\mathcal{D}$ the set of the predicate symbols for representing decisions.

In order to represent further knowledge about the domain under consideration, we will adopt also a set of predicate symbols for *beliefs*, denoted by $\mathcal{B}$. Furthermore, in many situations the knowledge about a decision making problem may be incomplete, and it may require to make assumptions to carry on the reasoning process. This will be tackled by selecting, in the set $\mathcal{B}$, those predicate symbols representing *presumptions* (denoted by $\mathcal{P}sm$). For instance, in the procurement example, the decision made by the buyer may (and will indeed) depend upon the way the buyer thinks the seller replies to the buyer's offer, either by accepting or by rejecting it. This can be represented by a presumption $\mathtt{reply}(x)$, where x is either `accept` or `reject`.

In a decision making problem, we need to express *preferences* between different goals and the *reservation value*, that is the lowest (in terms of preference) set of goals under which the agent cannot concede. For instance, in the procurement example, the buyer prefers to minimize the price. Hence, its knowledge base should somehow represent the fact that the goal `fast` should be preferred to `cheap`. On the other hand, the buyer is prepared to concede on the price in order to achieve an agreement with the seller, but it may be not ready to concede on the delivery time which must be low. Hence, its knowledge base should somehow represent the fact that these goals consist of its reservation value.

Finally, we allow the representation of explicit *incompatibilities* between goals and/or decisions. For instance, different alternatives for the same decision predicate are incompatible with each other, e.g. `s(a)` is incompatible with `s(b)`. On the other hand, different goals *may be* incompatible with one another. For instance, `cheap` is incompatible with `expensive`, whereas `expensive` is not incompatible with `good`. Incompatibilities between goals and between decisions will be represented through a binary relation denoted by $\mathcal{I}$.

The above informal discussion can be summarized by the definition of *decision framework* (Definition 8 below). For the sake of simplicity, in this definition, as well as in the rest of the paper, we will assume some familiarity with the basic notions of logic languages (such as terms, atomic formulae, clauses etc.) Moreover, we will not explicitly introduce formally all the components of the underlying logic language, in order to focus our attention to those components which are relevant to our decision making context. So, for instance, we assume that the constants and function symbols over which terms are built (i.e. predicate arguments) are given. Finally, given a set of predicate symbols X in the language, we will still use X to denote the set of all possible atomic formulae built on predicates belonging to X. If not clear from the context, we will point out whether we refer to the predicate symbols in X rather than to the atomic formulae built on X.

Definition 8 (*Decision framework*) A *decision framework* is a tuple $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$, where:

- $\mathcal{DL} = \mathcal{G} \cup \mathcal{D} \cup \mathcal{B}$ is a set of predicate symbols called the **decision language**, where we distinguish between goals ($\mathcal{G}$), decisions ($\mathcal{D}$) and beliefs ($\mathcal{B}$);
- $\mathcal{P}sm$ is a set of atomic formulae built upon predicates in $\mathcal{DL}$ called **presumptions**;
- $\mathcal{I}$ is the **incompatibility relation**, i.e. a binary relation over atomic formulae in $\mathcal{G}$, $\mathcal{B}$ or $\mathcal{D}$. $\mathcal{I}$ is not necessarily symmetric;
- $\mathcal{T}$ is a logic **theory** built upon $\mathcal{DL}$; statements in $\mathcal{T}$ are clauses, each of which has a distinguished *name*;
- $\mathcal{P} \subseteq \mathcal{G} \times \mathcal{G}$ is the **priority** relation, namely a transitive, irreflexive and asymmetric relation over atomic formulae in $\mathcal{G}$;
- $\mathcal{RV}$ is a set of literals built upon predicates in $\mathcal{G}$, called the **reservation value**.

Let us summarize the intuitive meaning of the various components of the framework. The language $\mathcal{DL}$ is composed by:

- the set of *goal* predicates, i.e. some predicate symbols which represent the features that a decision must exhibit;
- the set $\mathcal{D}$ of *decision* predicates, i.e. some predicate symbols which represent the actions which must be performed or not; different atoms built on $\mathcal{D}$ represent different *alternatives*;
- the set $\mathcal{B}$ of *beliefs*, i.e. some predicate symbols which represent epistemic statements;.

In this way, we can consider multiple objectives which may or not be fulfilled by a set of decisions under certain circumstances.

We explicitly distinguish *presumable* (respectively *non-presumable*) literals which can (respectively cannot) be assumed to hold, as long as there is no evidence to the contrary. Decisions as well as some beliefs can be assumed. In this way, $\mathtt{DF}$ can model the incompleteness of knowledge.

The most natural way to represent conflicts in our object language is by means of some forms of logical negation. We consider two types of negation, as usual, e.g., in extended logic programming, namely *strong negation* $\neg$ (also called *explicit* or *classical negation*), and *weak negation* $\sim$, also called *negation as failure*. As a consequence we will distinguish between strong literals, i.e. atomic formulae possibly preceded by strong negation, and weak literals, i.e. literals of the form $\sim L$, where L is a strong literal. The intuitive meaning of a strong literal $\neg L$ is "L is definitely not the case", while $\sim L$ intuitively means "There is no evidence that L is the case".

The set $\mathcal{I}$ of incompatibilities contains some *default* incompatibilities related to negation on the one hand, and to the nature of decision predicates on the other hand. Indeed, given an atom A, we have $A \, \mathcal{I} \, \neg A$ as well as $\neg A \, \mathcal{I} \, A$. Moreover, $L \, \mathcal{I} \sim L$, whatever L is, representing the intuition that L is evidence to the contrary of $\sim L$. Notice, however, that we do not have $\sim L \, \mathcal{I} \, L$, as in the spirit of weak negation. Other default incompatibilities are related to decisions, since different alternatives for the same decision predicate are incompatible with one another. Hence, $D(a_1) \, \mathcal{I} \, D(a_2)$ and $D(a_2) \, \mathcal{I} \, D(a_1)$, D being a decision predicate in $\mathcal{D}$, and a_1 and a_2 being different constants representing different[2] alternatives for D. Depending on the particular decision problem being represented by the framework, $\mathcal{I}$ may contain further non-default incompatibilities. For instance, we may have $g \, \mathcal{I} \, g'$, where g, g' are different goals (as `cheap` $\mathcal{I}$ `expensive` in the procurement example). To summarize, the incompatibility relation captures the conflicts, either default or domain dependent, amongst decisions, beliefs and goals.

The incompatibility relation can be easily lifted to set of sentences. We say that two sets of sentences Φ_1 and Φ_2 are *incompatible* (still denoted by $\Phi_1 \, \mathcal{I} \, \Phi_2$) iff there is a sentence ϕ_1 in Φ_1 and a sentence ϕ_2 in Φ_2 such that $\phi_1 \, \mathcal{I} \, \phi_2$.

A theory gathers the statements about the decision problem.

Definition 9 (*Theory*) A *theory* $\mathcal{T}$ is an extended logic program, *i.e* a finite set of rules R: $L_0 \leftarrow L_1, \ldots, L_j, \sim L_{j+1}, \ldots, \sim L_n$ with $n \geq 0$, each L_i (with $i \geq 0$) being a strong literal in $\mathcal{L}$. R, called the unique *name* of the rule, is an atomic formula of $\mathcal{L}$. All variables occurring in a rule are implicitly universally quantified over the whole rule. A rule with variables is a scheme standing for all its ground instances.

Considering a decision problem, we distinguish:

- *goal rules* of the form R: $G_0 \leftarrow G_1, \ldots, G_n$ with $n > 0$, where each G_i ($i \geq 0$) is a goal literal in $\mathcal{DL}$ (or its strong negation). According to this rule, the goal G_0 is promoted (or demoted) by the combination of the goal literals in the body;
- *epistemic rules* of the form R: $B_0 \leftarrow B_1, \ldots, B_n$ with $n \geq 0$, where each B_i ($i \geq 0$) is a belief literal of $\mathcal{DL}$. According to this rule, B_0 is true if the conditions $B_1, \ldots, B_n$ are satisfied;
- *decision rules* of the form R: $G \leftarrow D_1(a_1), \ldots, D_m(a_m), B_1, \ldots, B_n$ with $m \geq 1, n \geq 0$. The head of the rule is a goal (or its strong negation). The body includes a set of decision literals ($D_i(a_i) \in \mathcal{L}$) and a (possibly empty) set of belief literals. According to this rule, the goal is promoted (or demoted) by the decisions $\{D_1(a_1), \ldots, D_m(a_m)\}$, provided that the conditions $B_1, \ldots, B_n$ are satisfied.

For simplicity, we will assume that the names of rules are neither in the bodies nor in the head of the rules thus avoiding self-reference problems. Moreover, we assume

[2] Notice that in general a decision can be addressed by more than two alternatives.

that the elements in the body of rules are independent (the literals cannot be deduced from each other), the decisions do not influence the beliefs, and the decisions have no side effects.

Considering statements in the theory is not sufficient to make a decision. In order to evaluate the previous statements, other relevant pieces of information should be taken into account, such as the priority amongst goals. For this purpose, we consider the *priority* relation $\mathcal{P}$ over the goals in $\mathcal{G}$, which is transitive, irreflexive and asymmetric. $G_1 \mathcal{P} G_2$ can be read "G_1 has priority over G_2". There is no priority between G_1 and G_2, either because G_1 and G_2 are *ex æquo* (denoted $G_1 \simeq G_2$), or because G_1 and G_2 are not comparable. The priority corresponds to the relative importance of the goals as far as solving the decision problem is concerned. For instance, we can prefer a fast service rather than a cheap one. This preference can be captured by the priority. The reservation is the minimal set of goals which needs to be reached. The reservation value is the least favourable point at which one will accept a negotiated agreement. It would mean the bottom line that one would be prepared to concede.

In order to illustrate the previous notions, we provide here the decision framework related to the problem described in Sect. 3.1.

Example 3 (Decision framework) We consider the procurement example which is described in Sect. 3.1. The buyer's decision problem is captured by a *decision framework* $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ where:

- the decision language $\mathcal{DL}$ distinguishes
  - a set of **goals** $\mathcal{G} = \{\mathtt{cheap}, \mathtt{expensive}, \mathtt{fast}, \mathtt{slow}, \mathtt{bad}, \mathtt{good}\}$. This set of literals identifies various goals as the cost (cheap or expensive), the quality of service (good or bad) and the availability (slow or fast),
  - a set of **decisions** $\mathcal{D} = \{\mathtt{s}(x) \mid x \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}\}$. This set of literals identifies different alternatives,
  - a set of **beliefs**, i.e. a set of literals identifying various features $\mathtt{Price}(x, y)$, $\mathtt{Resolution}(x, y)$ and $\mathtt{DeliveryTime}(x, y)$ with $x \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $y \in \{\mathtt{high}, \mathtt{low}\}$ (which means that y is the level of a certain feature of x) and a set of literals identifying the possible replies of the responders $\{\mathtt{reply}(y) \mid y \in \{\mathtt{accept}, \mathtt{reject}\}\}$;
- the set of presumptions $\mathcal{P}sm$ contains the possible replies;
- the incompatibility relation $\mathcal{I}$ is trivially defined. In particular,

$$\mathtt{reply(accept)}\ \mathcal{I}\ \mathtt{reply(reject)},$$
$$\mathtt{reply(reject)}\ \mathcal{I}\ \mathtt{reply(accept)},\ \text{and}$$
$$\mathtt{s}(x)\ \mathcal{I}\ \mathtt{s}(y),\ \text{with}\ x \neq y$$
$$\mathtt{good}\ \mathcal{I}\ \mathtt{bad}, \mathtt{bad}\ \mathcal{I}\ \mathtt{good}, \mathtt{expensive}\ \mathcal{I}\ \mathtt{cheap}, \mathtt{cheap}\ \mathcal{I}\ \mathtt{expensive},$$
$$\mathtt{slow}\ \mathcal{I}\ \mathtt{fast}, \mathtt{fast}\ \mathcal{I}\ \mathtt{slow};$$

- the theory $\mathcal{T}$ (whatever the agent is the buyer or the seller) is the set of rules shown in Table 2;
- the preferences of the buyer in our example are such that:
  $\mathtt{fast}\mathcal{P}\mathtt{cheap}$ and $\mathtt{fast}\mathcal{P}\mathtt{good}$;

Table 2 The rules of the agents

$\mathtt{r}_{11}(x)$	$\mathtt{expensive} \leftarrow \mathtt{s}(x), \mathtt{Price}(x, \mathtt{high}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{r}_{12}(x)$	$\mathtt{cheap} \leftarrow \mathtt{s}(x), \mathtt{Price}(x, \mathtt{low}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{r}_{21}(x)$	$\mathtt{good} \leftarrow \mathtt{s}(x), \mathtt{Resolution}(x, \mathtt{high}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{r}_{22}(x)$	$\mathtt{bad} \leftarrow \mathtt{s}(x), \mathtt{Resolution}(x, \mathtt{low}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{r}_{31}(x)$	$\mathtt{fast} \leftarrow \mathtt{s}(x), \mathtt{DeliveryTime}(x, \mathtt{low}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{r}_{32}(x)$	$\mathtt{slow} \leftarrow \mathtt{s}(x), \mathtt{DeliveryTime}(x, \mathtt{high}), \mathtt{reply}(\mathtt{accept})$
$\mathtt{f}_{11}$	$\mathtt{Price}(\mathtt{a}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{12}$	$\mathtt{Resolution}(\mathtt{a}, \mathtt{low}) \leftarrow$
$\mathtt{f}_{13}$	$\mathtt{DeliveryTime}(\mathtt{a}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{21}$	$\mathtt{Price}(\mathtt{b}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{22}$	$\mathtt{Resolution}(\mathtt{b}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{23}$	$\mathtt{DeliveryTime}(\mathtt{b}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{31}$	$\mathtt{Price}(\mathtt{c}, \mathtt{high}) \leftarrow$
$\mathtt{f}_{32}$	$\mathtt{Resolution}(\mathtt{c}, \mathtt{low}) \leftarrow$
$\mathtt{f}_{33}$	$\mathtt{DeliveryTime}(\mathtt{c}, \mathtt{low}) \leftarrow$
$\mathtt{f}_{41}$	$\mathtt{Price}(\mathtt{d}, \mathtt{low}) \leftarrow$
$\mathtt{f}_{42}$	$\mathtt{Resolution}(\mathtt{d}, \mathtt{low}) \leftarrow$
$\mathtt{f}_{43}$	$\mathtt{DeliveryTime}(\mathtt{d}, \mathtt{low}) \leftarrow$

– The reservation value of the buyer is defined as: $\mathcal{RV} = \{\mathtt{fast}\}$. If the agent is the seller, then the reservation value is defined as : $\mathcal{RV} = \{\mathtt{expensive}\}$.

Our formalism allows to capture the incomplete representation of a decision problem with presumable beliefs. Arguments are built upon these incomplete statements.

3.3 Arguments

In order to turn the decision framework presented in the previous section into a concrete argumentation framework, we need first to define the notion of argument. Since we want that our AF not only suggests some decisions but also provides an intelligible explanation of them, we adopt a tree-like structure of arguments. We adopt here the tree-like structure for arguments proposed in (Vreeswijk 1997) and we extend it with presumptions on the missing information.

Informally, an argument is a deduction for a conclusion from a set of presumptions represented as a tree, with conclusion at the root and presumptions at the leaves. Nodes in this tree are connected by the inference rules, with sentences matching the head of an inference rule connected as parent nodes to sentences matching the body of the inference rule as children nodes. The leaves are either presumptions or the special extra-logical symbol $\top$, standing for an empty set of premises. Formally:

Definition 10 (*Structured argument*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. A *structured argument* built upon $\mathtt{DF}$ is composed by a conclusion, some premises, some presumptions, and some sentences. These elements are abbreviated by the corresponding prefixes (e.g. conc stands for conclusion). A structured argument $\bar{\mathrm{A}}$ can be:

1. a *hypothetical argument* built upon an unconditional ground statement. If L is either a decision literal or a presumable belief literal (or its strong/weak negation), then the argument built upon a ground instance of this presumable literal is defined as follows:

$$\begin{aligned}
&\mathtt{conc}(\bar{\mathrm{A}}) = L,\\
&\mathtt{premise}(\bar{\mathrm{A}}) = \emptyset,\\
&\mathtt{psm}(\bar{\mathrm{A}}) = \{L\},\\
&\mathtt{sent}(\bar{\mathrm{A}}) = \{L\}.
\end{aligned}$$

   or
2. a *built argument* built upon a rule such that all the literals in the body are the conclusion of arguments.
   (2.1) If f is a fact in $\mathcal{T}$ (i.e. $\mathtt{body}(f) = \top$[3]), then the *trivial* argument $\bar{\mathrm{A}}$ built upon this fact is defined as follows:

$$\begin{aligned}
&\mathtt{conc}(\bar{\mathrm{A}}) = \mathtt{head}(f),\\
&\mathtt{premise}(\bar{\mathrm{A}}) = \{\top\},\\
&\mathtt{psm}(\bar{\mathrm{A}}) = \emptyset,\\
&\mathtt{sent}(\bar{\mathrm{A}}) = \{\mathtt{head}(f)\}.
\end{aligned}$$

   (2.2) If r is a rule in $\mathcal{T}$ with $\mathtt{body}(r) = \{L_1, \ldots, L_j, \sim L_{j+1}, \ldots, \sim L_n\}$ and there is a collection of structured arguments $\{\bar{\mathrm{A}}_1, \ldots, \bar{\mathrm{A}}_n\}$ such that, for each strong literal $L_i \in \mathtt{body}(r)$, $L_i = \mathtt{conc}(\bar{\mathrm{A}}_i)$ with $i \leq j$ and for each weak literal $\sim L_i \in \mathtt{body}(r)$, $\sim L_i = \mathtt{conc}(\bar{\mathrm{A}}_i)$ with $i > j$, we define the *tree* argument $\bar{\mathrm{A}}$ built upon the rule r and the set $\{\bar{\mathrm{A}}_1, \ldots, \bar{\mathrm{A}}_n\}$ of structured arguments as follows:

$$\begin{aligned}
&\mathtt{conc}(\bar{\mathrm{A}}) = \mathtt{head}(r),\\
&\mathtt{premise}(\bar{\mathrm{A}}) = \mathtt{body}(r),\\
&\mathtt{psm}(\bar{\mathrm{A}}) = \bigcup_{\bar{\mathrm{A}}_i \in \{\bar{\mathrm{A}}_1, \ldots, \bar{\mathrm{A}}_n\}} \mathtt{psm}(\bar{\mathrm{A}}_i),\\
&\mathtt{sent}(\bar{\mathrm{A}}) = \mathtt{body}(r) \cup \{\mathtt{head}(r)\} \cup \bigcup_{\bar{\mathrm{A}}_i \in \{\bar{\mathrm{A}}_1, \ldots, \bar{\mathrm{A}}_n\}} \mathtt{sent}(\bar{\mathrm{A}}_i).
\end{aligned}$$

   The set of structured arguments $\{\bar{\mathrm{A}}_1, \ldots, \bar{\mathrm{A}}_n\}$ is denoted by $\mathtt{sbarg}(\bar{\mathrm{A}})$, and its elements are called the *subarguments* of $\bar{\mathrm{A}}$.

[3] $\top$ denotes the unconditionally true statement.

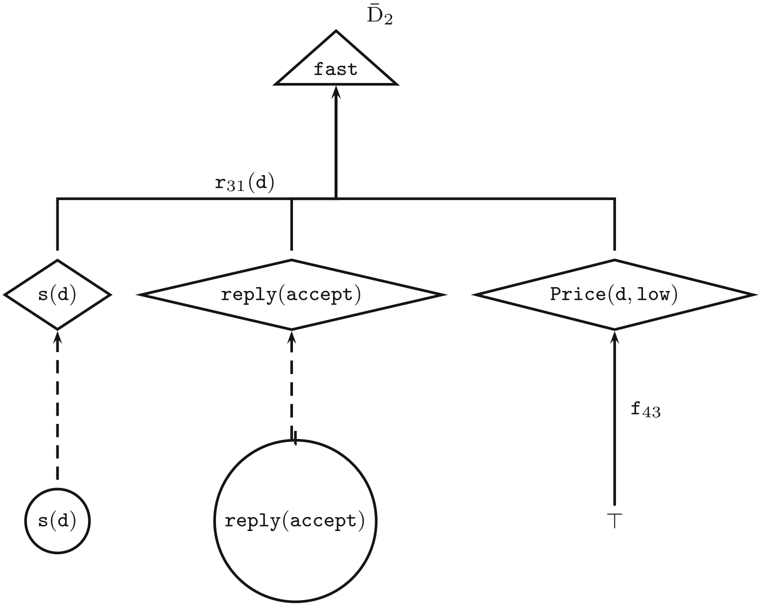


Fig. 1 The argument $\bar{D}_2$ concluding fast

The set of arguments built upon DF is denoted by $\mathcal{A}(\mathrm{DF})$.

Notice that the subarguments of a tree argument concluding the weak literals are hypothetical arguments. Indeed, the conclusion of an hypothetical argument could be a strong or a weak literal, while the conclusion of a built argument is a strong literal. As in (Vreeswijk 1997), we consider composite arguments, called *tree* arguments, and atomic arguments, called *trivial* arguments. Unlike the other definitions of arguments (set of assumptions, set of rules), our definition considers that the different premises can be challenged and can be supported by subarguments. In this way, arguments are intelligible explanations. Moreover, we consider *hypothetical* arguments which are built upon missing information or a suggestion, i.e. a decision. In this way, our framework allows to reason further by making suppositions related to the unknown beliefs and over possible decisions.

Let us consider the previous example.

Example 4 (Arguments) The arguments $\bar{D}_2$ and $\bar{C}$ concluding fast are depicted in Figs. 1 and 2, respectively. They are arguments concluding that the availability is promoted since the delivery time of the services c and d is low. For this purpose we need to suppose that the seller's reply will be an acceptance. An argument can be represented as a tree where the root is the conclusion (represented by a triangle) directly connected to the premises (represented by losanges) if they exist, and where leaves are either decisions/presumptions (represented by circles) or the unconditionally true statement. Each plain arrow corresponds to a rule (or a fact) where the head node corresponds to the head of the rule and the tall nodes represent the literals in the body of the rule. The tree arguments $\bar{C}$ and $\bar{D}_2$ are composed of three subarguments:

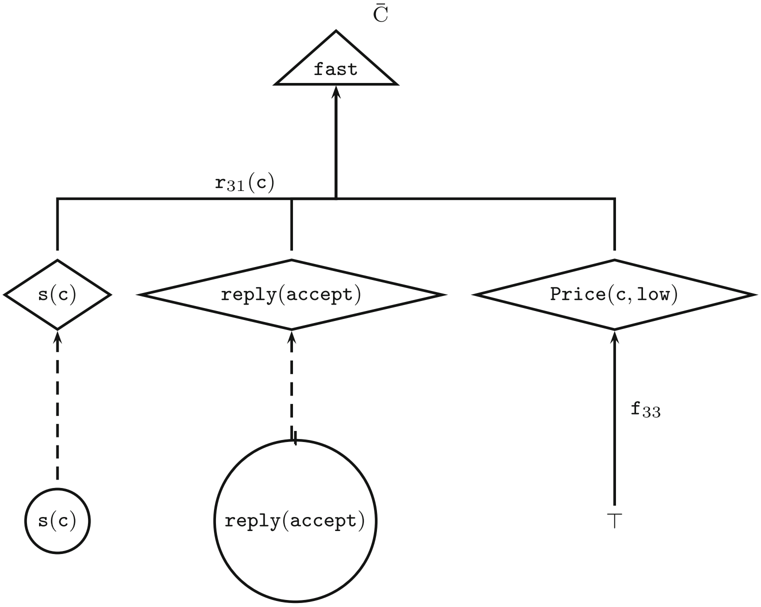


Fig. 2 The argument $\bar{C}$ concluding `fast`

two hypothetical and one trivial argument. Neither trivial arguments nor hypothetical arguments contain subarguments.

3.4 Interactions Between Arguments

In order to turn the decision framework into an argumentation framework, we need to capture the interactions between arguments. The interactions amongst structured arguments may come from their conflicts and from the priority over the goals which are promoted by these arguments. We examine in turn these different sources of interaction. Firstly, we define the attack relation amongst conflicting structured arguments in the same way we have defined the attack relation in the assumption-based argumentation frameworks. Secondly, we define the strength of arguments. Finally, we define the defeat relation amongst the structured arguments to capture the whole of interactions amongst them.

Since their sentences are conflicting, the structured arguments interact with one another. For this purpose, we define the following attack relation.

Definition 11 (*Attack relation*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, and $\bar{A}, \bar{B} \in \mathcal{A}(\mathtt{DF})$ be two structured arguments. $\bar{A}$ `attacks` $\bar{B}$ iff $\mathtt{sent}(\bar{A})\, \mathcal{I}\, \mathtt{sent}(\bar{B})$.

This relation encompasses both the direct (often called *rebuttal*) attack due to the incompatibility of the conclusions, and the indirect (often called *undermining*) attack, i.e. directed to a “subconclusion”. According to this definition, if an argument attacks a subargument, the whole argument is attacked.

Let us go back to our example.

Example 5 (Attack relation) $\bar{\mathrm{D}}_2$ (respectively $\bar{\mathrm{C}}$) is built upon the hypothetical subargument supposing $\mathtt{s(d)}$ (respectively $\mathtt{s(c)}$). Therefore, $\bar{\mathrm{C}}$ and $\bar{\mathrm{D}}_2$ attack each other.

Arguments are *concurrent* if their conclusions are identical or incompatible. In order to compare the strength of concurrent arguments, various domain-independent principles of commonsense reasoning can be applied. According to the specificity principle (Simari and Loui 1992), the most specific argument is the stronger one. According to the weakest link principle (Amgoud and Cayrol 2002), an argument cannot be justified unless all of its subarguments are justified. In accordance with the last link principle (Prakken and Sartor 1997), the strength of our arguments comes from the preferences between the sentence of the arguments. By contrast, the strength of our argument does not depend on the quality of information used to build that argument but it is determined by its conclusion.

Definition 12 (*Strength relation*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework and $\bar{\mathrm{A}}_1, \bar{\mathrm{A}}_2 \in \mathcal{A}(\mathtt{DF})$ be two structured and concurrent. $\bar{\mathrm{A}}_1$ is stronger than $\bar{\mathrm{A}}_2$ (denoted $\bar{\mathrm{A}}_1 \mathcal{P} \bar{\mathrm{A}}_2$) iff $\mathtt{conc}(\bar{\mathrm{A}}_1) = g_1 \in \mathcal{G}$, $\mathtt{conc}(\bar{\mathrm{A}}_2) = g_2 \in \mathcal{G}$ and $g_1 \mathcal{P} g_2$.

Due to the definition of $\mathcal{P}$ over $\mathcal{T}$, the relation $\mathcal{P}$ is transitive, irreflexive and asymmetric over $\mathcal{A}(\mathtt{DF})$.

The attack relation and the strength relation can be combined. As in (Amgoud and Cayrol 1998; Bench-Capon 2002), we distinguish between one argument attacking another, and that attack succeeding due to the strength of arguments.

Definition 13 (*Defeat relation*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework and $\bar{\mathrm{A}}$ and $\bar{\mathrm{B}}$ be two structured arguments. $\bar{\mathrm{A}}$ *defeats* $\bar{\mathrm{B}}$ iff:

1. $\bar{\mathrm{A}}$ $\mathtt{attacks}$ $\bar{\mathrm{B}}$;
2. and it is not the case that $\bar{\mathrm{B}} \mathcal{P} \bar{\mathrm{A}}$.

Similarly, we say that a set S of structured arguments defeats a structured argument $\bar{\mathrm{A}}$ if $\bar{\mathrm{A}}$ is defeated by one argument in S.

Let us consider our example.

Example 6 (Defeat relation) As previously mentioned, $\bar{\mathrm{C}}$ and $\bar{\mathrm{D}}_2$ attack each other and they conclude the same goal $\mathtt{fast}$. We can deduce that $\bar{\mathrm{C}}$ and $\bar{\mathrm{D}}_2$ defeat each other.

3.5 Argumentation Framework

We are now in the position of summarizing what is our argumentation framework for decision making. In doing this, we also inherit the semantics defined by Dung to analyse when a decision can be considered acceptable.

As we have seen, in our argumentation-based approach for decision making, arguments motivate decisions and they can be defeated by other arguments. More formally, our argumentation framework (AF for short) is defined as follows.

Definition 14 (*AF*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. The *argumentation framework for decision making* built upon $\mathtt{DF}$ is a pair $\mathtt{AF} = \langle \mathcal{A}(\mathtt{DF}), \mathtt{defeats} \rangle$ where $\mathcal{A}(\mathtt{DF})$ is the finite set of structured arguments built upon $\mathtt{DF}$ as defined in Definition 8, and $\mathtt{defeats} \subseteq \mathcal{A}(\mathtt{DF}) \times \mathcal{A}(\mathtt{DF})$ is the binary relation over $\mathcal{A}(\mathtt{DF})$ as defined in Definition 13.

We adapt Dung's extension-based semantics in order to analyse whenever a set of structured arguments can be considered as subjectively justified with respect to the preferences.

Definition 15 (*Semantics*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework and $\mathtt{AF} = \langle \mathcal{A}(\mathtt{DF}), \mathtt{defeats} \rangle$ be our argumentation framework for decision making. For $\bar{\mathrm{S}} \subseteq \mathcal{A}(\mathtt{DF})$ a set of structured arguments, we say that:

- $\bar{\mathrm{S}}$ is *subjectively conflict-free* iff $\forall \bar{\mathrm{A}}, \bar{\mathrm{B}} \in \bar{\mathrm{S}}$ it is not the case that $\bar{\mathrm{A}}$ defeats $\bar{\mathrm{B}}$;
- $\bar{\mathrm{S}}$ is *subjectively admissible* (s-admissible for short), denoted $\mathtt{sadm}_{\mathtt{AF}}(\bar{\mathrm{S}})$, iff $\bar{\mathrm{S}}$ is subjectively conflict-free and $\bar{\mathrm{S}}$ defeats every argument $\bar{\mathrm{A}}$ such that $\bar{\mathrm{A}}$ defeats some argument in $\bar{\mathrm{S}}$;

For simplicity, we restrict ourselves to the subjective admissibility, but the other Dung's extension-based semantics (cf Definition 2) can be easily adapted.

Formally, given a structured argument $\bar{\mathrm{A}}$, let

$$\mathtt{dec}(\bar{\mathrm{A}}) = \{D(a) \in \mathtt{psm}(\bar{\mathrm{A}}) \mid D \text{ is a decision predicate}\}$$

be the set of decisions supported by the structured argument $\bar{\mathrm{A}}$.

The decisions are *suggested* to reach a goal if they are supported by a structured argument concluding this goal and this argument is a member of an s-admissible set of arguments.

Definition 16 (*Credulous decisions*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $\mathtt{D} \subseteq \mathcal{D}$ be a set of decisions. The decisions $\mathtt{D}$ **credulously argue for** g iff there exists an argument $\bar{\mathrm{A}}$ in an s-admissible set of arguments such that $\mathtt{conc}(\bar{\mathrm{A}}) = g$ and $\mathtt{dec}(\bar{\mathrm{A}}) = \mathtt{D}$. We denote $\mathtt{val}_c(\mathtt{D})$ the set of goals in $\mathcal{G}$ for which the set of decisions $\mathtt{D}$ credulously argues.

It is worth noticing that the decisions which credulously argue for a goal cannot contain mutual exclusive alternatives for the same decision predicate. This is due to the fact that an s-admissible set of arguments is subjectively conflict-free.

If we consider the structured arguments $\bar{\mathrm{A}}$ and $\bar{\mathrm{B}}$ supporting the decisions $D(a)$ and $D(b)$ respectively where a and b are mutually exclusive alternatives, we have $D(a)\ \mathcal{I}\ D(b)$ and $D(b)\ \mathcal{I}\ D(a)$ and so, either $\bar{\mathrm{A}}$ $\mathtt{defeats}$ $\bar{\mathrm{B}}$ or $\bar{\mathrm{B}}$ $\mathtt{defeats}$ $\bar{\mathrm{A}}$ or both of them depending on the strength of these arguments.

Proposition 1 (Mutual exclusive alternatives) *Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $\mathtt{AF} = \langle \mathcal{A}(\mathtt{DF}), \mathtt{defeats} \rangle$ be the argumentation framework for decision making built upon $\mathtt{DF}$. If $\bar{S}$ be an s-admissible set of arguments such that, for some $\bar{A} \in \bar{S}$, $g = \mathtt{conc}(\bar{A})$ and $D(a) \in \mathtt{psm}(\bar{A})$, then $D(b) \in \mathtt{psm}(\bar{A})$ iff $a = b$.*

However, notice that mutual exclusive decisions can be suggested for the same goal through different s-admissible sets of arguments. This case reflects the credulous nature of our semantics.

Definition 17 (*Skeptical decisions*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $\mathtt{D}, \mathtt{D}' \subseteq \mathcal{D}$ be two sets of decisions. The set $\mathtt{D}$ of decisions **skeptically argue for** g iff for all s-admissible set of arguments $\bar{\mathrm{S}}$ such that for some arguments $\bar{\mathrm{A}}$ in $\bar{\mathrm{S}}$ $\mathtt{conc}(\bar{\mathrm{A}}) = g$, then $\mathtt{dec}(\bar{\mathrm{A}}) = \mathtt{D}$. We denote $\mathtt{val}_s(\mathtt{D})$ the set of goals in $\mathcal{G}$ for which the set of decisions $\mathtt{D}$ skeptically argues. The decisions $\mathtt{D}$ is **skeptically preferred** to the decisions $\mathtt{D}'$ iff $\mathtt{val}_s(\mathtt{D})\mathcal{P}\mathtt{val}_s(\mathtt{D}')$.

Due to the uncertainties, some decisions satisfy goals for sure if they skeptically argue for them, or some decisions can possibly satisfy goals if they credulously argue for them. While the first case is required for convincing a risk-averse agent, the second case is enough to convince a risk-taking agent. Since some ultimatum choices amongst various justified sets of alternatives are not always possible, we will consider in this paper the most "skeptically preferred" decisions.

The decision making process can be described as the cognitive process in which an agent evaluates the alternatives that are available, according to their features, to determine whether and how they satisfy his needs. The principle for decision making we adopt is that higher-ranked goals should be pursued at the expense of lower-ranked goals, and thus choices enforcing higher-ranked goals should be preferred to those enforcing lower-ranked goals. We are in a situation where there is a ranking of individual objects (the preferences between goals) and we need a ranking that involve subsets of these objects (See Barber et al. 2004 for a survey). For this purpose, we adopt the minmax ordering.

Definition 18 (*Preferences*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. We consider $\mathtt{G}$, $\mathtt{G}'$ two sets of goals in $\mathcal{G}$ and $\mathtt{D}$, $\mathtt{D}'$ two sets of decisions in $\mathcal{D}$. $\mathtt{G}$ is **preferred** to $\mathtt{G}$ (denoted $\mathtt{G}\mathcal{P}\mathtt{G}'$) iff

1. $\mathtt{G} \supset \mathtt{G}'$, and
2. $\forall g \in \mathtt{G} \setminus \mathtt{G}'$ there is no $g' \in \mathtt{G}'$ such that $g'\mathcal{P}g$.

$\mathtt{D}$ is **credulously preferred** (respectively **skeptically preferred**) to $\mathtt{D}'$ (denoted $\mathtt{D}\mathcal{P}_c\mathtt{D}'$ and $\mathtt{D}\mathcal{P}_s\mathtt{D}'$) iff $\mathtt{val}_c(\mathtt{D})\mathcal{P}\mathtt{val}_c(\mathtt{D}')$ (respectively $\mathtt{val}_s(\mathtt{D})\mathcal{P}\mathtt{val}_s(\mathtt{D}')$).

Formally, let

$$\begin{aligned}\mathcal{SAD} = \{\mathtt{D} \mid \mathtt{D} \subseteq \mathcal{D} \text{ such that } \mathcal{RV} \subseteq \mathtt{val}_s(\mathtt{D}) \text{ and}\\ \forall \mathtt{D}' \subseteq \mathcal{D} \text{ it is not the case that } \mathcal{RV} \subseteq \mathtt{val}_s(\mathtt{D}') \text{ and } \mathtt{val}_s(\mathtt{D}')\ \mathcal{P}\ \mathtt{val}_s(\mathtt{D})\}\end{aligned}$$

be the set of decisions which can be skeptically accepted by the agent. Additionally, let

$$\mathcal{SAG} = \{\mathtt{G} \mid \mathtt{G} \subseteq \mathcal{G} \text{ such that } \mathtt{G} = \mathtt{val}_s(\mathtt{D}) \text{ with } \mathtt{D} \in \mathcal{SAD}\}$$

be the goals which can be skeptically reached by the agent.

As an example of the decision making principle, consider the goals g_0, g_1 and g_2 such that $g_2 \mathcal{P} g_1$, $g_2 \mathcal{P} g_0$ and $\mathcal{RV} = \{g_0\}$. $\{g_2, g_1, g_0\}$ is preferred to both $\{g_2, g_0\}$, $\{g_2, g_1\}$ whereas $\{g_2, g_0\}$, $\{g_2, g_1\}$ are incomparable and so equally preferred. However, $\{g_2, g_1\}$ cannot be reached by the agent since it does not include the reservation value.

Let us consider now the buyer's decision problem in the procurement example.

Example 7 (Semantics) The structured argument $\bar{\mathrm{C}}$ and $\bar{\mathrm{D}}_2$, which are depicted in Figs. 1 and 2, conclude `fast`. Actually, the sets of decisions {`s(c)`} and {`s(d)`} credulously argue for *fast*. The decisions {`s(d)`} skeptically argue for `cheap` and *a fortiori* credulously argue for it. Therefore, {`s(d)`} is a skeptically acceptable set of decisions. The reservation value of the buyer only contains `fast`. Therefore, {`s(d)`} is skeptically preferred to {`s(c)`} and {`s(d)`} is a skeptical acceptable set of decisions due to the reservation value and the priority over the goals.

In our example, there is only one suggested set of decisions.

Since agents can consider multiple objectives which may not be fulfilled all together by a set of non-conflicting decisions, they may have to make some concessions, i.e. surrender previous proposals. Concessions are crucial features of agent-based negotiation. Rosenschein and Zlotkin have proposed a monotonic concession protocol for bilateral negotiations in Rosenschein and Zlotkin (1994). In this protocol, each agent starts from the deal that is best for him and either concedes or stands stills in each round. A (monotonic) concession means that an agent proposes a new deal that is better for the other agent. Differently from Rosenschein and Zlotkin (1994), we do not assume that the agent has an interlocutor and if it does, that it does not know the preferences of its interlocutors. We say that a decision is a minimal concession whenever there is no other preferred decisions.

Definition 19 (*Minimal concession*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. The decision $\mathtt{dec} \in \mathcal{D}$ is a **concession** with respect to $\mathtt{dec}' \in \mathcal{D}$ iff there exists a set of decisions $\mathtt{D}$ such that $\mathtt{dec} \in \mathtt{D}$ and for all $\mathtt{D}' \subseteq \mathcal{D}$ with $\mathtt{dec}' \in \mathtt{D}'$, it is not the case that $\mathtt{D}\mathcal{P}\mathtt{D}'$. The decision $\mathtt{dec}$ is a **minimal concession** wrt $\mathtt{dec}'$ iff it is a concession wrt $\mathtt{dec}'$ and there is no $\mathtt{dec}'' \in \mathcal{D}$ such that

- $\mathtt{dec}''$ is a concession wrt $\mathtt{dec}'$, and
- there is $\mathtt{D}'' \subseteq \mathcal{D}$ with $\mathtt{dec}'' \in \mathtt{D}''$ with $\mathtt{D}''\mathcal{P}\mathtt{D}$.

The minimal concessions are computed by the computational counterpart of our argumentation framework.

Example 8 (Minimal concession) According to the buyer, {`s(c)`} is a minimal concession with respect to {`s(d)`}.

3.6 Computational Counterpart

Having defined our argumentation framework for decision making, we need to find a computational counterpart for it. For this purpose, we move our AF to an ABF (cf. Sect. 2) which can be computed by the dialectical proof procedure of (Dung et al. 2006) extended in (Gartner and Toni 2007). So that, we can compute the suggestions for reaching a goal. Additionally, we provide the mechanism for solving a decision problem, modeling the intuition that high-ranked goals are preferred to low-ranked goals which can be withdrawn.

The idea is to map our argumentation framework built upon a decision framework into a collection of assumption-based argumentation frameworks, that we call *practical assumption-based argumentation frameworks* (PABFs for short). Basically, for each rule r in the theory we consider the assumption $\sim \mathtt{deleted}(r)$ in the set of possible assumptions. By means of this new predicate, we distinguish in a PABF the several distinct arguments that give rise to the same conclusion. Considering a set of goals, we allow each PABF in the collection to include (or not) the rules whose heads are these goals (or their strong negations). Indeed, two practical assumption-based frameworks in this collection may differ in the set of rules that they adopt. In this way, the mechanism consists of a search in the collection of PABFs.

Definition 20 (*PABF*) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \ \mathcal{I}\ , \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework and $\mathtt{G} \in \mathcal{G}$ a set of goals such that $\mathtt{G} \supseteq \mathcal{RV}$. A *practical assumption-based argumentation framework built upon* DF *associated with the goals* G is a tuple $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}) = \langle \mathcal{L}_{\mathtt{DF}}, \mathcal{R}_{\mathtt{DF}}, \mathcal{A}sm_{\mathtt{DF}}, \mathcal{C}on_{\mathtt{DF}} \rangle$ where:

(i) $\mathcal{L}_{\mathtt{DF}} = \mathcal{DL} \cup \{\mathtt{deleted}\}$;[4]

(ii) $\mathcal{R}_{\mathtt{DF}}$, the set of inference rules, is defined as follows:
- For each rule $r \in \mathcal{T}$, there exists an inference rule $R \in \mathcal{R}_{\mathtt{DF}}$ such that $\mathtt{head}(R) = \mathtt{head}(r)$ and $\mathtt{body}(R) = \mathtt{body}(r) \cup \{\sim \mathtt{deleted}(r)\}$;
- If $\mathtt{r}_1, \mathtt{r}_2 \in \mathcal{T}$ with $\mathtt{head}(\mathtt{r}_1)\ \mathcal{I}\ \mathtt{head}(\mathtt{r}_2)$ and it is not the case that $\mathtt{head}(\mathtt{r}_2) \mathcal{P} \mathtt{head}(\mathtt{r}_1)$, then the inference rule

$$\mathtt{deleted}(\mathtt{r}_2) \leftarrow \sim \mathtt{deleted}(\mathtt{r}_1) \text{ is in } \mathcal{R}_{\mathtt{DF}}.$$

(iii) $\mathcal{A}sm_{\mathtt{DF}}$, the set of assumptions, is defined such that $\mathcal{A}sm_{\mathtt{DF}} = \Delta \cup \Phi \cup \Psi \cup \Upsilon \cup \Sigma$ where:
- $\Delta = \{D(a) \in \mathcal{L} \mid D(a) \text{ is a decision literal }\}$,
- $\Phi = \{B \in \mathcal{B} \mid B \in \mathcal{P}sm\}$,
- $\Psi = \{\sim \mathtt{deleted}(r) \mid r \in \mathcal{T} \text{ and } \mathtt{head}(r) \in \{L, \neg L\} \text{ s.t. } L \notin \mathcal{G}\}$,
- $\Upsilon = \{\sim \mathtt{deleted}(r) \mid r \in \mathcal{T} \text{ and } \mathtt{head}(r) \in \{g, \neg g\} \text{ s.t. } g \in \mathcal{RV}\}$;
- $\Sigma = \{\sim \mathtt{deleted}(r) \mid r \in \mathcal{T} \text{ and } \mathtt{head}(r) \in \{g, \neg g\} \text{ s.t. } g \in \mathtt{G} - \mathcal{RV}\}$;

(iv) $\mathcal{C}on_{\mathtt{DF}}$ the set of contraries is defined such that for all $\alpha \in \mathcal{A}sm_{\mathtt{DF}}$, $y \in \mathcal{C}on(\alpha)$ iff $y\ \mathcal{I}\ \alpha$.

The set of practical assumption-based argumentation frameworks built upon DF and associated with the goals $\mathtt{G}'$ with $\mathcal{RV} \subseteq \mathtt{G}' \subseteq \mathtt{G}$ will be denoted $\mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$.

Case (i) defines the language. In order to capture the decision problem within an assumption-based argumentation framework, we have extended the decision language to include a predicate symbol deleted, which is used to specify whether or not a rule is adopted within the PABF. It is worth noticing that the definition of arguments in the

[4] We assume $\mathtt{deleted} \notin \mathcal{L}$.

ABF (cf Definition 6) focuses attention on the candidate assumptions and ignores the internal structure of arguments. In order to distinguish in a PABF the several distinct arguments that give rise to the same conclusion, we have named the rules used to deduce it. Therefore, an argument in a PABF contains a set of assumptions of the following schemata $\sim \mathtt{deleted}(r)$, for all rule r used by the argument.

Case (ii) defines the inference rules. Firstly, there is an inference rule for each rule of the theory. For this purpose, the body of each rule r is extended by adding the assumption $\sim \mathtt{deleted}(r)$. Referring to Example 3, the rule $\mathtt{r}_{11}(x)$ becomes

$$\mathtt{expensive} \leftarrow \mathtt{s}(x), \mathtt{Price}(x, \mathtt{high}), \mathtt{reply}(\mathtt{accept}), \sim \mathtt{deleted}(r_{11}(x)).$$

In this way, the assumption $\sim \mathtt{deleted}(\mathtt{r}_{11}(x))$ allows an argument to use this rule.

Secondly, the inference rules include not only the original deduction rules but also the conflicts amongst the rules having incompatible heads. It is worth noticing that the attack relation between arguments in the ABF (cf Definition 7) ignores the possible conflicts amongst the heads of rules which are not assumptions. In order to capture these conflicts, we have introduced rules which allow the defeasibility of rules. Referring to the example, we introduce, e.g.,

$$\mathtt{deleted}(\mathtt{r}_{12}(x)) \leftarrow\sim \mathtt{deleted}(\mathtt{r}_{11}(x))$$

modeling the given incompatibility $\mathtt{cheap}\,\mathcal{I}\,\mathtt{expensive}$. Obviously, we also introduce,

$$\mathtt{deleted}(\mathtt{r}_{11}(x)) \leftarrow\sim \mathtt{deleted}(\mathtt{r}_{12}(x))$$

modeling the given incompatibility $\mathtt{expensive}\,\mathcal{I}\,\mathtt{cheap}$. Our treatment of conflicting rules requires not to interfere with our treatment of priorities which is inspired by (Kowalski and Toni 1996). Referring to the example, we introduce, e.g.,

$$\mathtt{deleted}(\mathtt{r}_{12}(x)) \leftarrow\sim \mathtt{deleted}(\mathtt{r}_{31}(x))$$

modeling the given priority $\mathtt{cheap}\mathcal{P}\mathtt{fast}$. In this way, the corresponding literal $\sim \mathtt{deleted}(\mathtt{r}_{31}(x))$ must be assumed in order to handle this priority. Obviously, we do not introduce,

$$\mathtt{deleted}(\mathtt{r}_{31}(x)) \leftarrow\sim \mathtt{deleted}(\mathtt{r}_{12}(x)).$$

Case (iii) defines the assumptions. The decisions are obviously possible assumptions. In the same way, a PABF adopts a presumable belief if this is a presumption of the corresponding $\mathtt{AF}$. Referring to the example, an argument, which assumes that the reply is an acceptance, can be built since $\mathtt{reply}(\mathtt{accept}) \in \mathcal{A}sm_{\mathtt{DF}}$. Each framework adopts the epistemic rules, i.e r with $\mathtt{head}(r) \in \{L, \neg L\}$ and $L \notin \mathcal{G}$, by having the assumption $\sim \mathtt{deleted}(r)$ in its set of assumptions.

We want to go through the set of goals such that high-ranked goals are preferred to low-ranked goals and the reservation value is the minimal set of goals we want to reach.

For this purpose, we adopt the rules concluding the goals (or their negation) in the reservation value, i.e r with $\texttt{head}(r) \in \{g, \neg g\}$ and $g \in \mathcal{RV}$, by having the assumption $\sim \texttt{deleted}(r)$ in its set of assumptions. However, each framework in $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$ can or cannot adopt the rules concluding the goals (or their negation) which are not in the reservation value, i.e. r with $\texttt{head}(r) \in \{g, \neg g\}$ and $g \in \texttt{G} - \mathcal{RV}$ by having the assumption $\sim \texttt{deleted}(r)$ in its set of assumptions. Referring to the running example and considering the goal $\texttt{cheap}$ the strongest structured arguments concluding $\texttt{cheap}$, requires $\texttt{r}_{12}(x)$ to be built within the PABF if $\sim \texttt{deleted}(\texttt{r}_{12}(x)) \in \mathcal{A}sm_{\texttt{DF}}$.

Case (iv) defines the contrary relation of a PABF which trivially comes from the incompatibility relation and which comes from the contradiction of $\texttt{deleted}(r)$ with $\sim \texttt{deleted}(r)$ whatever the rule r is.

Arguments will be built upon rules, the candidate decisions, and by making suppositions within the presumable beliefs. Formally, given a decision framework $\texttt{DF}$ and a practical assumption-based framework

$$\texttt{pabf}_{\texttt{DF}}(\texttt{G}) = \langle \mathcal{L}_{\texttt{DF}}, \mathcal{R}_{\texttt{DF}}, \mathcal{A}sm_{\texttt{DF}}, \mathcal{C}on_{\texttt{DF}} \rangle, \text{ we define}$$
$$\Sigma = \{\sim \texttt{deleted}(r) \in \mathcal{A}sm_{\texttt{DF}} \mid r \in \mathcal{T} \text{ and } \texttt{head}(r) \in \{g, \neg g\} \text{ and } g \in \texttt{G} - \mathcal{RV}\}$$

as the set of goal rules considered in this PABF.

The practical assumption-based argumentation frameworks built upon a decision framework and associated with some goals include (or not) the rules concluding these goals which are more or less prior. This allows us to associate the set $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$ with a priority relation, denoted $\mathcal{P}$, modeling the intuition that, in solving a decision problem, high-ranked goals are preferred to low-ranked goals.

Definition 21 (*Priority over PABF*) Let $\texttt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, $\texttt{G} \in \mathcal{G}$ a set of goals such that $\texttt{G} \supseteq \mathcal{RV}$ and $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$ be the set of PABFs associated with the goals $\texttt{G}$.

$$\forall \texttt{G}_1, \texttt{G}_2 \text{ such that } \mathcal{RV} \subseteq \texttt{G}_1, \texttt{G}_2 \subseteq \texttt{G}\ \forall \texttt{pabf}_{\texttt{DF}}(\texttt{G}_1), \texttt{pabf}_{\texttt{DF}}(\texttt{G}_2) \in \texttt{PABFS}_{\texttt{DF}}(\texttt{G}),$$
$$\texttt{pabf}_{\texttt{DF}}(\texttt{G}_1) \mathcal{P} \texttt{pabf}_{\texttt{DF}}(\texttt{G}_2) \text{ iff :}$$

- $\texttt{G}_1 \supset \texttt{G}_2$, and
- $\forall g_1 \in \texttt{G}_1 \setminus \texttt{G}_2$ there is no $g_2 \in \texttt{G}_2$ such that $g_2 \mathcal{P} g_1$.

Due to the properties of set inclusion, the priority relation $\mathcal{P}$ is transitive, irreflexive and asymmetric over $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$.

In order to illustrate the previous notions, let us go back to our example.

Example 9 (PABF) Given the decision framework (cf example 3) capturing the decision problem of the buyer ($\mathcal{RV} = \{\texttt{fast}\}$). We consider the set of goals $\{\texttt{fast}, \texttt{cheap}, \texttt{good}\}$. We denote this set $\texttt{G}$. We will consider the collection of practical assumption-based argumentation frameworks $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$.

Let $\texttt{pabf}_{\texttt{DF}}(\texttt{G}) = \langle \mathcal{L}_{\texttt{DF}}, \mathcal{R}_{\texttt{DF}}, \mathcal{A}sm_{\texttt{DF}}, \mathcal{C}on_{\texttt{DF}} \rangle$ be a practical assumption-based argumentation framework in $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$. This PABF is defined as follows:

- $\mathcal{L}_{\texttt{DF}} = \mathcal{DL} \cup \{\texttt{deleted}\}$, where $\mathcal{DL}$ is defined as in the previous example and $\texttt{deleted}$ specifies if a rule does not hold;

- $R_{\mathtt{DF}}$ is defined by the rules in Table 3;
- $\mathcal{A}sm_{\mathtt{DF}} = \Delta \cup \Gamma \cup \Upsilon \cup \Sigma$ where:
  - $\Delta = \{\mathtt{s}(x) \mid x \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}\}$,
  - $\Phi = \{\mathtt{reply}(y) \mid y \in \{\mathtt{accept}, \mathtt{reject}\}\}$,
  - $\Psi = \{\sim \mathtt{deleted}(\mathtt{f}_{11}), \sim \mathtt{deleted}(\mathtt{f}_{12}), \sim \mathtt{deleted}(\mathtt{f}_{13}) \sim \mathtt{deleted}(\mathtt{f}_{21}), \sim \mathtt{deleted}(\mathtt{f}_{22}), \sim \mathtt{deleted}(\mathtt{f}_{23}), \sim \mathtt{deleted}(\mathtt{f}_{31}), \sim \mathtt{deleted}(\mathtt{f}_{32}), \sim \mathtt{deleted}(\mathtt{f}_{33}), \sim \mathtt{deleted}(\mathtt{f}_{41}), \sim \mathtt{deleted}(\mathtt{f}_{42}), \sim \mathtt{deleted}(\mathtt{f}_{43})\}$,
  - $\Upsilon = \{\sim \mathtt{deleted}(\mathtt{r}_{31})(x), \sim \mathtt{deleted}(\mathtt{r}_{32})(x)\}$,
  - $\Sigma \subseteq \{\sim \mathtt{deleted}(r_{11}(x)), \sim \mathtt{deleted}(r_{12}(x)), \sim \mathtt{deleted}(r_{21}(x)), \sim \mathtt{deleted}(r_{22}(x))\}$;
- $\mathcal{C}on_{\mathtt{DF}}$ is defined trivially. In particular,

$$\mathcal{C}on(\mathtt{s}(x)) = \{\mathtt{s}(y) \mid y \neq x\},$$

  for each r, $\mathtt{deleted}(r) \in \mathcal{C}on(\sim \mathtt{deleted}(r))$ if $\sim \mathtt{deleted}(r) \in \mathcal{A}sm_{\mathtt{DF}}$.

The possible sets Σ considered for the definition of the practical assumption-based argumentation framework $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_i) \in \mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$ (with $1 \leq i \leq 6$) are such that:

- $\mathtt{G}_1 = \{\mathtt{cheap}, \mathtt{good}, \mathtt{fast}\}$ with

$$\begin{aligned}\Sigma_1 = \{&\sim \mathtt{deleted}(r_{11}(x)), \sim \mathtt{deleted}(r_{12}(x)),\\ &\sim \mathtt{deleted}(r_{21}(x)), \sim \mathtt{deleted}(r_{22}(x)),\\ &\sim \mathtt{deleted}(r_{31}(x)), \sim \mathtt{deleted}(r_{32}(x))\};\end{aligned}$$

- $\mathtt{G}_2 = \{\mathtt{good}, \mathtt{fast}\}$ with

$$\begin{aligned}\Sigma_2 = \{&\sim \mathtt{deleted}(r_{21}(x)), \sim \mathtt{deleted}(r_{22}(x)),\\ &\sim \mathtt{deleted}(r_{31}(x)), \sim \mathtt{deleted}(r_{32}(x))\};\end{aligned}$$

- $\mathtt{G}_3 = \{\mathtt{good}, \mathtt{fast}\}$ with

$$\begin{aligned}\Sigma_3 = \{&\sim \mathtt{deleted}(r_{21}(x)), \sim \mathtt{deleted}(r_{22}(x)),\\ &\sim \mathtt{deleted}(r_{31}(x)), \sim \mathtt{deleted}(r_{32}(x))\};\end{aligned}$$

- $\mathtt{G}_4 = \{\mathtt{fast}\}$ with

$$\Sigma_4 = \{\sim \mathtt{deleted}(r_{31}(x)), \sim \mathtt{deleted}(r_{32}(x))\}.$$

It is clear that $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_1)\mathcal{P}\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_2)$, $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_1)\mathcal{P}\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_3)$, $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_2) \mathcal{P}\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_4)$ and $\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_3)\mathcal{P}\mathtt{pabf}_{\mathtt{DF}}(\mathtt{G}_4)$.

Having defined the PABFs, we show how a structured argument as in Definition 10 corresponds to an argument in one of the PABFs. To do this, we first define a mapping between a structured argument and a set of assumptions.

Table 3 The rules of the PABF

$\texttt{expensive} \leftarrow \texttt{s}(x), \texttt{Price}(x, \texttt{high}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{11}(x))$

$\texttt{cheap} \leftarrow \texttt{s}(x), \texttt{Price}(x, \texttt{low}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{12}(x))$

$\texttt{good} \leftarrow \texttt{s}(x), \texttt{Resolution}(x, \texttt{high}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{21}(x))$

$\texttt{bad} \leftarrow \texttt{s}(x), \texttt{Resolution}(x, \texttt{low}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{22}(x))$

$\texttt{fast} \leftarrow \texttt{s}(x), \texttt{DeliveryTime}(x, \texttt{low}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{31}(x))$

$\texttt{slow} \leftarrow \texttt{s}(x), \texttt{DeliveryTime}(x, \texttt{high}), \texttt{reply(accept)}, \sim \texttt{deleted}(r_{32}(x))$

$\texttt{Price(a, high)} \leftarrow \sim \texttt{deleted}(f_{11})$

$\texttt{Resolution(a, low)} \leftarrow \sim \texttt{deleted}(f_{12})$

$\texttt{DeliveryTime(a, high)} \leftarrow \sim \texttt{deleted}(f_{13})$

$\texttt{Price(b, high)} \leftarrow \sim \texttt{deleted}(f_{21})$

$\texttt{Resolution(b, high)} \leftarrow \sim \texttt{deleted}(f_{22})$

$\texttt{DeliveryTime(b, high)} \leftarrow \sim \texttt{deleted}(f_{23})$

$\texttt{Price(c, high)} \leftarrow \sim \texttt{deleted}(f_{31})$

$\texttt{Resolution(c, low)} \leftarrow \sim \texttt{deleted}(f_{32})$

$\texttt{DeliveryTime(c, low)} \leftarrow \sim \texttt{deleted}(f_{33})$

$\texttt{Price(d, low)} \leftarrow \sim \texttt{deleted}(f_{41})$

$\texttt{Resolution(d, low)} \leftarrow \sim \texttt{deleted}(f_{42})$

$\texttt{DeliveryTime(d, low)} \leftarrow \sim \texttt{deleted}(f_{43})$

Definition 22 (*Mapping between arguments*) Let $\texttt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. Let $\bar{\text{A}}$ be a structured argument in $\mathcal{A}(\texttt{DF})$ and concluding $\alpha \in \mathcal{DL}$. The corresponding set of assumptions deducing α (denoted $\ell(\bar{\text{A}})$) is defined according to the nature of $\bar{\text{A}}$.

- If $\bar{\text{A}}$ is a hypothetical argument, then $\ell(\bar{\text{A}}) = \{\alpha\}$.
- If $\bar{\text{A}}$ is a trivial argument built upon the fact f, then $\ell(\bar{\text{A}}) = \{\sim \texttt{deleted}(f)\}$.
- If $\bar{\text{A}}$ is a tree argument, then $\ell(\bar{\text{A}}) = \{\sim \texttt{deleted}(r_1), \ldots, \sim \texttt{deleted}(r_n)\} \cup \{L_1, \ldots, L_m\}$ where:
  (i) $r_1, \ldots, r_n$ are the rules of $\bar{\text{A}}$;
  (ii) the literals $L_1, \ldots, L_m$ are the presumptions and the decision literals of $\bar{\text{A}}$.

The mapping is materialized through a bijection ℓ: $\mathcal{A}(\texttt{DF}) \rightarrow \mathcal{A}sm_{\texttt{DF}}$ where $\mathcal{A}sm_{\texttt{DF}}$ is the set of possible assumptions of one of the PABFs built upon $\texttt{DF}$ and $\mathcal{A}(\texttt{DF})$ is the set of structured arguments built upon $\texttt{DF}$. If $\bar{\text{S}}$ is a set of arguments $\mathcal{A}(\texttt{DF})$, we denote $\ell(\bar{\text{S}})$ the corresponding set of assumptions. Formally,

$$\ell(\bar{\text{S}}) = \{\ell(\bar{\text{A}}) \mid \bar{\text{A}} \in \bar{\text{S}}\}$$

There is a one-to-one mapping between arguments in our AF and arguments in some corresponding PABFs.

Lemma 1 (Mapping between arguments) *Let* $\texttt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ *be a decision framework,* $\texttt{G} \in \mathcal{G}$ *a set of goals and* $\texttt{PABFS}_{\texttt{DF}}(\texttt{G})$ *be the set of PABFs associated with the goals* $\texttt{G}$.

1. *Given a structured argument built upon* $\mathtt{DF}$ *concluding* $\alpha \in \mathcal{DL}$*, there is a corresponding argument deducing* α *in some PABFs of* $\mathtt{PABFS}(G)$.
2. *Given an atomic formula* $\alpha \in \mathcal{DL}$ *and an argument of a PABF in* $\mathtt{PABFS}(G)$ *deducing* α*, there exists a corresponding structured argument in* $\mathcal{A}(\mathtt{DF})$ *concluding* α.

Let us consider the previous example.

Example 10 (Assumptions) The arguments in some PABFs corresponding to the structured arguments $\bar{\mathrm{D}}_2$ and $\bar{\mathrm{C}}$ include the following set of assumptions:

- $\ell(\bar{\mathrm{D}}_2) = \{\sim \mathtt{deleted}(\mathtt{r}_{31}(\mathtt{d})), \sim \mathtt{deleted}(\mathtt{f}_{43}), \mathtt{s}(\mathtt{d}), \mathtt{reply}(\mathtt{accept})\}$;
- $\ell(\bar{\mathrm{C}}) = \{\sim \mathtt{deleted}(\mathtt{r}_{31}(\mathtt{c})), \sim \mathtt{deleted}(\mathtt{f}_{33}), \mathtt{s}(\mathtt{c}), \mathtt{reply}(\mathtt{accept})\}$;

Both of them are tree argument. The corresponding set of assumptions $\ell(\bar{\mathrm{D}}_2)$ considers the literals $\sim \mathtt{deleted}(r_{31}(\mathtt{d}))$ and $\sim \mathtt{deleted}(\mathtt{f}_{43})$ since $\bar{\mathrm{D}}_2$ is built upon these rules. Moreover, the literal $\mathtt{s}(\mathtt{d})$ (respectively $\mathtt{reply}(\mathtt{accept})$) is a decision literal (respectively a presumption).

In order to compute our extension-based semantics, we explore the collection of PABFs associated to our AF in order to find the PABF which deduces the strongest goals as possible. Indeed, we have developed a mechanism to explore the collection of PABFs associated to our AF in order to compute it. If an s-admissible set of structured arguments concludes some goals, then there is a corresponding admissible set of assumptions in one of the corresponding PABFs and there is no other PABF, where an admissible set of assumptions deduces stronger goals.

Theorem 1 (Mapping between semantics) *Let* $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ *be a decision framework and* $\mathtt{AF} = \langle \mathcal{A}(\mathtt{DF}), \mathtt{defeats} \rangle$ *be our argumentation framework for decision making. Let us consider* $G \in \mathcal{G}$ *with* $G \supseteq \mathcal{RV}$.

- *If there is an s-admissible set of structured arguments* $\bar{S}_1$ *concluding* G_1 *with* $\mathcal{RV} \subseteq G_1 \subseteq G$ *such there is no s-admissible set of structured arguments concluding* G_2 *with* $\mathcal{RV} \subseteq G_2 \subseteq G$ *with* $G_2 \mathcal{P} G_1$*, then there is* $\mathtt{pabf}_1 \in \mathtt{PABFS}_{\mathtt{DF}}(G)$ *such that the corresponding set of assumptions* $\ell(\bar{S}_1)$ *is admissible within* $\mathtt{pabf}_1$ *and there is no* $\mathtt{pabf}_2 \in \mathtt{PABFS}_{\mathtt{DF}}(G)$*, with* $\mathtt{pabf}_2 \mathcal{P} \mathtt{pabf}_1$*, which contains an admissible set of assumptions deducing* G_2.
- *If there is* $\mathtt{pabf}_1 \in \mathtt{PABFS}_{\mathtt{DF}}(G)$ *which contains an admissible set of assumptions* A_1 *deducing* G_1 *with* $\mathcal{RV} \subseteq G_1 \subseteq G$ *such that there is no* $\mathtt{pabf}_2 \in \mathtt{PABFS}_{\mathtt{DF}}(G)$*, with* $\mathtt{pabf}_2 \mathcal{P} \mathtt{pabf}_1$*, which contains an admissible set of assumptions deducing* G_2 *with* $\mathcal{RV} \subseteq G_2 \subseteq G$ *and* $G_2 \mathcal{P} G_1$*, then the corresponding structured arguments* $\ell^{-1}(A_1)$ *concluding* G_1 *is in a s-admissible set and there is no other structured arguments* $\bar{S}_2$ *concluding* G_2 *which is in an s-admissible set.*

4 Implementation

The implementation of our framework is called MARGO. We describe here its usage in particular in the context of service-oriented agents. MARGO stands for Multiattribute

ARGumentation framework for Opinion explanation. MARGO is written in Prolog and available in GPL (GNU General Public License) at http://margo.sourceforge.net/.

In order to be computed by MARGO, the file, which describes the decision problem, contains:

- a set of decisions, i.e. some lists which contain the alternatives courses of actions;
- possibly a set of incompatibilities, i.e. some couples such that the first component is incompatible with the second component;
- possibly a set of symmetric incompatibilities, i.e. some couples such that the first component is incompatible with the second component and conversely;
- a set of decisions rules, i.e. some triples of name—head–body which are simple Prolog representations of the decision rules in our AF;
- possibly a set of goal rules, i.e. some triples of name—head–body which are simple Prolog representations of the goal rules in our AF;
- possibly a set of epistemic rules, i.e. some triples of name—head–body which are simple Prolog representations of the epistemic rules in our AF;
- possibly a set of priorities, some couples of goals such that the former have priority over the latter;
- a set of presumable belief literals;
- a reservation value, i.e. a list which contains the minimal set of goals which needs to be reached.

We can note that the incompatibilities between the mutual exclusive alternatives are implicit in the MARGO language. It is worth noticing that MARGO attempts to narrow the gap between the specification of the decision framework and the corresponding code.

The main predicate `admissible(+G,?AG,?AD)` succeeds when `AG` are the acceptable goals extracted form `G` and `AD` are the acceptable decisions. The predicate for argument manipulation `admissibleArgument(+C,?P,?S)` succeeds when `P` are the premises and `S` are the presumptions of an argument deriving the conclusion `C` and this argument is in a subjectively admissible set.

Example 11 (Usage) Table 4 presents our example, as described in Sect. 3.2, in the MARGO syntax. `admissible([cheap, fast, good], AG, AD)` returns:

```
AG = [cheap,  fast]
AD = [s(d)]
```

`admissibleArgument(cheap,P,S)` returns:

```
P = [s(c), price(c,low), reply(accept)]
S = [s(c),reply(accept)]
```

MARGO has been used for service composition and orchestration within the ARGUGRID project.[5] As discussed in (Toni et al. 2008), the ArguGRID system contains a semantic composition environment, allowing users to interact with their agents,

[5] http://www.argugrid.eu.

Table 4 The decision problem of the buyer in the MARGO syntax

```
decision([s(X)]).
decisionrule(r11(X), expensive,
                [s(X), price(X,high), reply(accept)]).
decisionrule(r12(X), cheap,
                [s(X), price(X,low), reply(accept)]).
decisionrule(r21(X), good,
                [s(X), resolution(X,high), reply(accept)]).
decisionrule(r22(X), bad,
                [s(X), resolution(X,low),  reply(accept)]).
decisionrule(r31(X), fast,
                [s(X), deliveryTime(X,low), reply(accept)]).
decisionrule(r32(X), slow,
                [s(X), deliveryTime(X,high), reply(accept)]).
epistemicrule(f11, price(a,high), []).
epistemicrule(f12, resolution(a,low), []).
epistemicrule(f13, deliveryTime(a,high), []).
epistemicrule(f21, price(b,high), []).
epistemicrule(f22, resolution(b,high), []).
epistemicrule(f23, deliveryTime(b,high), []).
epistemicrule(f31, price(c,high), []).
epistemicrule(f32, resolution(c,low), []).
epistemicrule(f33, deliveryTime(c,low), []).
epistemicrule(f41, price(d,low), []).
epistemicrule(f42, resolution(d,low), []).
epistemicrule(f43, deliveryTime(d,low), []).
presumable(reply(accept)).
presumable(reply(reject)).
priority(fast, cheap).
priority(fast, good).
rv([fast]).
sincompatibility(fast,slow).
sincompatibility(cheap,expensive).
sincompatibility(good,bad).
```

and a grid middleware for the actual deployment of services. Service-oriented computing is an interesting test bed for multi-agent system techniques, where agents need to adopt a variety of roles that will empower them to provide services in open and distributed systems. Moreover, service-oriented computing can benefit from multi-agent systems technologies by adopting the coordination mechanisms, interaction protocols, and decision-making tools designed for multi-agent systems, e.g. MARGO.

Bromuri et al. (2009) have demonstrated the use of a fully decentralised multi-agent system supporting agent-automated service discovery, agent-automated service selection, and agent-automated negotiation of Service Level Agreements (SLAs) for the selected services.

5 Negotiation

Requester agents select services according to their suitability to fulfil high-level user requirements. These agents use argumentation in order to assess suitability and identify "optimal" services. They argue internally using our concrete argumentation system linking decisions on selecting services, (a possibly incomplete description of) the features of these services, the benefits that these features guarantee (under possibly incomplete knowledge). The ArguGRID system uses the MARGO tool for multi-attribute qualitative decision-making to support the decision on suitable services. As soon as the requester agents identify a suitable service, it engages in a negotiation process with the provider agent for that service.

The negotiation aims at agreeing a SLA on the usage of the identified service. While one of the agent starts by asserting a first proposal, the other agent replies with a counter-proposal. An agent must adopt one of these attitudes: (i) either it **stands still**, i.e. it repeats its previous proposal; (ii) or it **concedes**, i.e. it withdraws to put forward one of its previous proposal and it considers another one. In order to articulate these attitudes, the negotiation is conducted using a realisation of the minimal concession strategy of (Morge and Mancarella 2010). This strategy consists of adhering the reciprocity principle during the negotiation. If the interlocutor stands still, then the agent will stand still. Whenever the interlocutor has made a concession, it will reciprocate by conceding as well. It is worth noticing that the third step in the negotiation has a special status, in that the agent has to concede. If the agent is not able to concede (e.g. there is no other service which satisfies its constraints), the agent will standstill. If an acceptable offer has been put forward by the interlocutor, the player accepts it. When the player can no more concede, it stops the negotiation. It is worth noticing that contrary to Dung et al. (2008), our strategy does not stop the negotiation after 3 consecutive standstills but the strategy allows to concede after them.

Due to the finiteness assumption of the language, and hence the finiteness of possible decisions, the dialogue is also finite.

Proposition 1 (Terminaison) *The negotiations are finite.*

Due to the finiteness assumption and the definition of the minimal concession strategy over the potential agreements, it is not difficult to see that such negotiations are successful, if a potential agreement exists.

Proposition 2 (Success) *If both players adopt a minimal concession strategy and a potential agreement exists, then the negotiation is a success.*

Since a player will concede at a certain point even if its interlocutor stands still since it can no more concede, the negotiation between two players adopting the minimal concession strategy go throw the whole sets of acceptable services. In other words, our realisation of the minimal concession strategy allows to reach an agreement. However, this realisation of the minimal concession strategy is not in a pure symmetric Nash equilibrium. It means that when an agent is adopting the minimal concession strategy, the other agent can do better than using this strategy. Differently from Dung et al. (2008), our realisation of the minimal strategy allows to reach an agreement even if the agents do not know the preferences and the reservation value of the other agents. However, this realisation of the minimal concession strategy is not in a pure symmetric Nash equilibrium.

The final agreement of the negotiation is said to be a Pareto optimal if it is not possible to strictly improve the individual welfare of an agent without making the other worse off. This is the case of our realisation of the minimal concession strategy in a one-to-one bargaining.

Theorem 2 (Pareto optimal) *If both players adopt a minimal concession strategy and a potential agreement exists, then the outcome of the dialogue is Pareto optimal.*

The outcome is Pareto optimal since the concessions are minimal.

6 Related Works

Unlike the theoretical reasoning, practical reasoning is not only about whether some beliefs are true, but also about whether some actions should or should not be performed. The practical reasoning (Raz 1978) follows three main steps: (i) *deliberation*, i.e. the generation of goals; (ii) *means-end reasoning*, i.e. the generation of plans; (iii) *decision-making*, i.e. the selection of plans that will be performed to reach the selected goals.

Argumentation has been put forward as a promising approach to support decision making (Fox and Parsons 1997). While influence diagrams and belief networks (Oliver and Smith 1988) require that all the factors relevant for a decision are identified *a priori*, arguments are defeasible or reinstantiated in the light of new information not previously available.

Amgoud and Prade (2009) present a general and abstract argumentation framework for multi-criteria decision making which captures the mental states (goals, beliefs and preferences) of the decision makers. For this purpose, the arguments prescribe actions to reach goals if theses actions are feasible under certain circumstances. These arguments, eventually conflicting, are balanced according to their strengths. Our specific and concrete argumentation framework is in conformance with this approach. The argumentation-based decision making process envisaged by Amgoud and Prade (2009) is split in different steps where the arguments are successively constructed, weighted, confronted and evaluated. By contrast, our computation interleaves the construction of arguments, the construction of counterarguments, the evaluation of the generated arguments and the determination of concessions. Moreover, our argumentation-based decision process suggests some decisions even if low-ranked goals cannot be reached.

Bench-Capon and Prakken (2006) formalize defeasible argumentation for practical reasoning. As in Amgoud and Prade (2009), they select the best course of actions by confronting and evaluating arguments. Bench-Capon and Prakken focus on the abductive nature of practical reasoning which is directly modelled within in our framework.

Kakas and Moraitis (2003) propose an argumentation-based framework for decision making of autonomous agents. For this purpose, the knowledge of the agent is split and localized in different modules representing different capabilities. As Bench-Capon and Prakken (2006) and Amgoud and Prade (2009), their framework is a particular instantiation of the abstract argumentation (Dung 1995). Whereas Kakas and Moraitis (2003) is committed to one argumentation semantics, we can deploy our framework to several semantics by relying on assumption-based argumentation.

Rahwan et al. (2003) distinguish different approaches for automated negotiation, including game-theoretic approaches (e.g Rosenschein and Zlotkin 1994), heuristicbased approaches (e.g. Faratin et al. 1998) and argumentation-based approaches (e.g. Amgoud et al. 2007; Bench-Capon and Prakken 2006; Kakas and Moraitis 2003) which allow for more sophisticated forms of interaction. By adopting the argumentation-based approach of negotiation, agents deal naturally with new information in order to mutually influence their behaviors. Indeed, the two first approaches do not allow agents for exchanging opinions about offers. By arguing (even if it is internally), agents can take into account the information given by its interlocutors in a negotiation process (eg. rejecting some offers). Moreover, the agents can make some concessions. In this perspective, Amgoud et al. (2007) propose a general framework for argumentation-based negotiation. They define formally the notions of concession, compromise and optimal solution. Our argumentation-based mechanism for decision making can be used for exploiting such a feature.

Finally, to the best of our knowledge, few implementation of argumentation over actions exist. CaSAPI[6] (Gartner and Toni 2007) and DeLP[7] (García and Simari 2004) are restricted to the theoretical reasoning. PARMENIDES[8] (Atkinson et al. 2006) is a software to structure the debate over actions by adopting a particular argumentation scheme. GORGIAS[9] (Demetriou and Kakas 2003) implements an argumentation-based framework to support the decision making of an agent within a modular architecture. Like the latter, our implementation, called MARGO, incorporates abduction on missing information. Moreover, we can easily extend it to compute the competing semantics since MARGO is built upon CaSAPI which is an argumentation engine that implements the dispute derivations described in Dung et al. (2007).

7 Discussion

To our best knowledge, our argumentation-based mechanism for decision-making is the only concrete argumentation system allowing concessions which is a crucial feature for negotiations. Our framework is built upon assumption-based argumentation frameworks, and provides mechanisms to evaluate decisions, to suggest decisions, and to interactively explain in an intelligible way the choice which has been made to make a certain decision, along with the concessions, if any, made to support this choice. The underlying language in which all the components of a decision problem are represented is a logic-based language, in which preferences can be attached to goals. In our framework, arguments are defined by means of tree-structures, thus facilitating their intelligibility. The concession-based mechanism is a crucial feature of our framework required in different applications such as service selection or agent-based negotiation. Our framework has been implemented and actually exploited in different application domains, such as agent-based negotiation (Morge and Mancarella 2010; Bromuri et al. 2009), service-oriented agents (Guo et al. 2009), resource

[6] http://www.doc.ic.ac.uk/~dg00/casapi.html.

[7] http://lidia.cs.uns.edu.ar/DeLP.

[8] http://cgi.csc.liv.ac.uk/~katie/Parmenides.html.

[9] http://www.cs.ucy.ac.cy/~nkd/gorgias/.

allocation (Morge et al. 2009), computational model of trust (Matt et al. 2010) or embodied conversational agents (Morge et al. 2010).

Our preliminar negotiation model we have considered in this paper only allows the exchange of proposals and counter-proposals. That is the reason why each agent decides on its own goals and priorities. This negotiation model have been extended in Morge et al. (2013) for exchanging, generating and evaluating arguments during negotiations. The extra information carried out by these arguments allows agents to influence other agents' preference model, and so it allows to decrease the number messages required to reach an agreement. However, this negotiation model can only handle negotiation about fixed item/service. In future works, we want to apply our argumentation-based mechanism for integrative negotiations rather than distributive negotiations. Contrary to distributive negotiations, all aspects are considered for a solution that maximizes the social welfare, such as new services to accommodate each other's needs for a better deal. We aim at adopting this negotiation model and extend the strategy to generate and evaluate additional sub-items.

Acknowledgments We would like to thank the anonymous reviewers for their detailed comments on this paper.

Appendix: Proofs

This appendix includes the proofs considered in this paper.

Proof 1 (Mapping between arguments) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV}\rangle$ be a decision framework, $\mathtt{G} \in \mathcal{G}$ a set of goals and $\mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$ be the set of PABFs associated with the goals $\mathtt{G} \supseteq \mathcal{RV}$.

1. Let $\bar{\mathrm{A}}$ be a structured argument concluding $\alpha \in \mathcal{DL}$. The corresponding argument in some of the PABFs (denoted $\ell(\bar{\mathrm{A}})$) is defined in Definition 22.
2. Let us consider α an atomic formula in $\mathcal{DL}$ and $\mathtt{a}\colon \mathtt{A} \vdash \alpha$ be an argument within one PABF in $\mathtt{PABFS}_{\mathtt{DF}}(L)$.
   - Either $\alpha \in \mathcal{A}sm_{\mathtt{DF}}$. Therefore, α is deduced by the singleton $\{\alpha\}$ and $\ell^{-1}(\mathtt{A}) = \bar{\mathrm{A}}$ is a hypothetical argument defined such that:

$$\begin{aligned}
&\mathtt{conc}(\bar{\mathrm{A}}) = \alpha,\\
&\mathtt{premise}(\bar{\mathrm{A}}) = \emptyset,\\
&\mathtt{psm}(\bar{\mathrm{A}}) = \{\alpha\},\\
&\mathtt{sent}(\bar{\mathrm{A}}) = \{\alpha\}.
\end{aligned}$$

   - Or $\alpha \notin \mathcal{A}sm_{\mathtt{DF}}$. Therefore, α is deduced by the set of premises $\mathtt{A}$ which, by definition, are all assumptions in $\mathcal{A}sm_{\mathtt{DF}}$ and, by construction, there is only one rule r with $\sim \mathtt{deleted}(r) \in \mathtt{A}$ such as $\mathtt{head}(r) = \alpha$.

- Either r is a fact of $\mathcal{T}$ deducing α and so, $\ell^{-1}(\mathtt{A}) = \bar{\mathrm{A}}$ is a trivial argument defined such that:

$$\begin{aligned} &\mathtt{conc}(\bar{\mathrm{A}}) = \alpha,\\ &\mathtt{premise}(\bar{\mathrm{A}}) = \{\top\},\\ &\mathtt{psm}(\bar{\mathrm{A}}) = \emptyset,\\ &\mathtt{sent}(\bar{\mathrm{A}}) = \{\alpha\}. \end{aligned}$$

- Otherwise r is a rule of $\mathcal{T}$ deducing α with a non-empty body and so, $\ell^{-1}(\mathtt{A}) = \bar{\mathrm{A}}$ is a tree argument built upon the subarguments $\mathtt{sbarg}(\bar{\mathrm{A}})$ defined such that:

$$\begin{aligned} &\mathtt{conc}(\bar{\mathrm{A}}) = \alpha,\\ &\mathtt{premise}(\bar{\mathrm{A}}) = \mathtt{body}(r),\\ &\mathtt{psm}(\bar{\mathrm{A}}) = \cup_{\bar{\mathrm{A}}' \in \mathtt{sbarg}(\bar{\mathrm{A}})} \mathtt{psm}(\bar{\mathrm{A}}'),\\ &\mathtt{sent}(\bar{\mathrm{A}}) = \mathtt{body}(r) \cup \{\mathtt{head}(r)\} \cup \bigcup_{\bar{\mathrm{A}}' \in \mathtt{sbarg}(\bar{\mathrm{A}})} \mathtt{sent}(\bar{\mathrm{A}}'). \end{aligned}$$

Proof 2 (Mapping between semantics) Let $\mathtt{DF} = \langle \mathcal{DL}, \mathcal{P}sm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework and $\mathtt{AF} = \langle \mathcal{A}(\mathtt{DF}), \mathtt{defeats} \rangle$ be our argumentation framework for decision making. Let us consider $\mathtt{G} \in \mathcal{G}$ with $\mathtt{G} \supseteq \mathcal{RV}$.

- Let us consider an s-admissible set of structured arguments $\bar{\mathrm{S}}_1$ concluding $\mathtt{G}_1 \subseteq \mathtt{G}$. Due to the Lemma 1, we can built the set of arguments $\mathtt{S}_1$ such that for any structured argument $\bar{\mathrm{A}}_1 \in \bar{\mathrm{S}}_1$ there is an argument $\mathtt{a}_1 \in \mathtt{S}_1$, where $\mathtt{a}_1 : \ell(\bar{\mathrm{A}}_1) \vdash L$ is in some $\mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$. We consider here $\mathtt{pabf}^1_{\mathtt{DF}}(\mathtt{G}_1) \in \mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$ where all the arguments appear. Due to the construction of $\mathtt{S}_1$, the set of arguments $\mathcal{A}(\ell(\bar{\mathrm{S}}_1))$ is conflict-free and defend itself within $\mathtt{pabf}^1_{\mathtt{DF}}(G)$. Therefore, $\mathtt{S}_1$ is an admissible set. Let us consider a different $\mathtt{pabf}^2_{\mathtt{DF}}(\mathtt{G}_2) \in \mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$ such that $\mathtt{pabf}^2_{\mathtt{DF}}(\mathtt{G}_2)\mathcal{P}\mathtt{pabf}^1_{\mathtt{DF}}(\mathtt{G}_1)$. Due to the Definition 21, $\mathtt{G}_2 \supseteq \mathtt{G}_1$ and $\forall g_2 \in \mathtt{G}_2 \setminus \mathtt{G}_1$ there is no $g_1 \in \mathtt{G}_1$ such that $g_1 \mathcal{P} g_2$. If we suppose that $\mathtt{pabf}^2_{\mathtt{DF}}(\mathcal{G})$ contains an admissible set of arguments deducing $\mathtt{G}_2$, then the corresponding set of structured arguments concluding $\mathtt{G}_2$ is admissible. It is not the case.
- Let us consider $\mathtt{pabf}_1 \in \mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$ which contains an admissible set of assumptions $\mathtt{A}_1$ deducing $\mathtt{G}_1$ with $\mathcal{RV} \subseteq \mathtt{G}_1 \subseteq \mathtt{G}$. If we suppose that there is no $\mathtt{pabf}_2 \in \mathtt{PABFS}_{\mathtt{DF}}(\mathtt{G})$, with $\mathtt{pabf}_2 \mathcal{P} \mathtt{pabf}_1$, which contains an admissible set of assumptions deducing $\mathtt{G}_2 \subseteq \mathtt{G}$ with $\mathtt{G}_2 \mathcal{P} \mathtt{G}_1$, then the corresponding s-admissible set of structured arguments $\ell^{-1}(\mathtt{A}_1)$ concludes $\mathtt{G}_1$ and there is no other s-admissible set of structured arguments $\bar{\mathrm{S}}_2$ concluding $\mathtt{G}_2$.

Proof 3 (Pareto optimal) Let us consider a negotiation where an agent a plays the minimal concession strategy and the other agent b plays another strategy. If a potential agreement exists and the negotiation is a failure, the outcome is worst, in particular from the viewpoint of b, than the game where both a and b play a minimal concession strategy. Otherwise the negotiation is a success. If b did not concede at a certain point even if it was possible for him, then the outcome is worst from the viewpoint of a. If b has conceded every time it was possible for him and its concessions were not minimal the outcome is worst from the viewpoint of b.

References

Amgoud L, Cayrol C (1998) On the acceptability of arguments in preference-based argumentation. In: Proceedings of UAI, Madison, Wisconsin, USA. Morgan Kaufmann, Los Altos, pp 1–7

Amgoud L, Cayrol C (2002) A reasoning model based on the production of acceptable arguments. Ann Math AI 34(1–3):197–215

Amgoud L, Henri P (2009) Using arguments for making and explaining decisions. Artif Intell J 173(3–4):413–436

Amgoud L, Dimopoulos Y, Moraitis P (2007) A unified and general framework for argumentation-based negotiation. In: Proceedings of AAMAS, Honolulu, Hawaii, pp 963–970

Atkinson K, Bench-Capon T, McBurney P (2006) Computational representation of practical argument. Synthese (Special Issue Knowl Ration Action) 152(2):157–206

Barber S, Bossert W, Attanaik PK (2004) In: Barberà S, Hammond PJ, Seidl C (eds) Ranking sets of objects. Handbook of utility theory, vol 2. Kluwer, Dordrecht, pp 893–977

Bench-Capon TJM (2002) Value based argumentation frameworks. In: Proceedings of the 2nd international workshop on non-monotonic reasoning (NMR), pp 444–453

Bench-Capon T, Prakken H (2006) Justifying actions by accruing arguments. In: Proceedings of the 1st international conference on computational models of argument. IOS Press, Amsterdam, pp 247–258

Bondarenko A, Toni F, Kowalski R (1993) An assumption-based framework for non-monotonic reasoning. In: Proceedings of LPNMR. MIT Press, Cambridge

Bromuri S, Urovi V, Morge M, Toni F, Stathis K (2009) A multi-agent system for service discovery, selection and negotiation (demonstration). In: 8th international joint conference on autonomous agents and multiagent systems (AAMAS 2009), Budapest, Hungary, May 10–15, 2009, vol 2. IFAAMAS, 2009, pp 1395–1396. ISBN 978-0-9817381-7-8.

Chesñevar CI, Maguitman AG, Loui RP (2000) Logical models of argument. ACM Comput Surv 32(4):337–383. ISSN 0360–0300. doi:10.1145/371578.371581

Demetriou N, Kakas AC (2003) Argumentation with abduction. In: Proceedings of the 4th panhellenic symposium on logic

Dung PM (1995) On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif Intell 77(2):321–357. ISSN 0004–3702

Dung PM, Kowalski RA (2006) Dialectic proof procedures for assumption-based, admissible argumentation. Artif Intell 170(2):114–159

Dung PM, Mancarella P, Toni F (2007) Computing ideal sceptical argumentation. Artif Intell (Special Issue Argumentation) 171(10–15):642–674

Dung PM, Thang PM, Toni F (2008) Towards argumentation-based contract negotiation. In: Proceedings of the 2nd second international conference on computational models of argument. IOS Press, Amsterdam, pp 134–146

Faratin P, Sierra C (1998) Negotiation decision functions for autonomous agents. Int J Rob Auton Syst 24:3–4

Fox J, Parsons S (1997) Argumentation and decision making. In: Proceedings of IEE colliquium on decision making and problem solving, London

García AJ, Simari GR (2004) Defeasible logic programming: an argumentative approach. Theory Pract Log Program 4(1–2):95–138

Gartner D, Toni F (2007) CaSAPI: a system for credulous and sceptical argumentation. In: Proceedings of ArgNMR, pp 80–95

Guo L, Ghanem M, Curcin V, Azam N (2009) Intelligent agents in the service-oriented world—an industrial experience report. Web Intell Intell Agent Technol IEEE/WIC/ACM Int Conf 1:693–696. doi:10.1109/WI-IAT.2009.365

Jennings NR, Faratin P, Lomuscio AR, Parsons S, Sierra C, Wooldridge M (2001) Automated negotiation: prospects, methods and challenges. Int J Group Decis Negot 10(2):199–215

Kakas A, Moraitis P (2003) Argumentative-based decision-making for autonomous agents. In: Proceedings of AAMAS. ACM Press, New York, pp 883–890

Kowalski R, Toni F (1996) Abstract argumentation. Artif Intell Law J (Special Issue Log Models Argumentation) 4(3–4):275–296

Matt P-A, Morge M, Toni F (2010) Combining statistics and arguments to compute trust. In: van der Hoek W, Kaminka G, Lesprance Y, Luck M, Sen S (eds) Proceedings of the ninth international joint conference on autonomous agents and multi-agent systems (AAMAS'2010). International Foundation for Autonomous Agents and Multiagent Systems, pp 209–210

Morge M, Mancarella P (2010) Assumption-based argumentation for the minimal concession strategy. In: McBurney P, Rahwan I, Parsons S, Maudet N (eds) Argumentation in multi-agent systems, 6th international workshop, ArgMAS 2009, Budapest, Hungary, May 12, 2009. Revised selected and invited papers, volume 6057 of lecture notes in computer science. Springer, Berlin, pp 114–133. ISBN 978-3-642-12804-2. http://www.springerlink.com/content/902635985hw042n8/

Morge M, Abdel-Naby S, Beaufils B (2010) Towards a dialectical approach for conversational agents in selling situations. In: Rahwan I, McBurney P, Parsons S (eds) The seventh proceedings of the seventh international workshop on argumentation in multi-agent systems (ArgMAS'2010). International Foundation for Autonomous Agents and Multiagent Systems, pp 127–144

Morge M, Mathieu P, Mancarella P (2009) Assumption-based argumentation for the minimal concession strategy of agents engaged in resource negotiation. In: Actes des Cinquièmes Journées Francophones sur les Modèles Formels de l'Interaction (MFI'09), pp 232–242. ISBN 978-2-9527630-2-8

Morge M, McGinnis J, Bromuri S, Mancarella P, Stathis K, Toni F (2013) Multiagent systems & applications, volume 45 of studies in computational intelligence, chapter Argumentative agents for service-oriented computing. Springer, Berlin, pp 217–255

Oliver RM, Smith JQ (eds) (1988) Influence diagrams, belief nets and decision analysis. Wiley, New York

Prakken H, Sartor G (1997) Argument-based logic programming with defeasible priorities. J Appl Nonclassical Log 7:25–75

Rahwan I, Ramchurn SD, Jennings NR, McBurney P, Parsons S, Sonenberg L (2003) Argumentation-based negotiation. Knowl Eng Rev 18(4):343–375

Raz J (ed) (1978) Practical reasoning. Oxford University Press, Oxford

Rosenschein JS, Gilad Z (1994) Rules of encounter: designing conventions for automated negotiation among Computers. The MIT press series of artificial intelligence. MIT press, Cambridge

Simari GR, Loui RP (1992) A mathematical treatment of defeasible reasoning and its implementation. Artif Intell 53(2–3):125–157

Stournaras T (eds) (2007) Concrete scenarios identification and simple use cases. Deliverable document D1.1 ARGUGRID

Toni F, Grammatikou M, Kafetzoglou S, Lymberopoulos L, Papavassileiou S, Gaertner D, Morge M, Bromuri S, McGinnis J, Stathis K, Curcin V, Ghanem M, Guo L (2008) The argugrid platform: an overview. In: Proceedings of the 5th international workshop on grid economics and business models (GECON'08), volume 5206 of lecture notes in artificial intelligence, Las Palmas de Gran Canaria, Spain. Springer, Berlin, pp 217–225

Vreeswijk G (1997) Abstract argumentation systems. Artif Intell 90(1–2):225–279