An Argumentation-Based Approach to Modeling Decision Support Contexts with What-If Capabilities

Pietro Baroni and **Federico Cerutti** and **Massimiliano Giacomin** and **Giovanni Guida**
Dipartimento di Elettronica per l'Automazione, Università di Brescia,
Via Branze 38, I-25123 Brescia, Italy

Abstract

This paper describes a preliminary proposal of an argumentation-based approach to modeling articulated decision support contexts. The proposed approach encompasses a variety of argument and attack schemes aimed at representing basic knowledge and reasoning patterns for decision support. Some of the defined attack schemes involve attacks directed towards other attacks, which are not allowed in traditional argumentation frameworks but turn out to be useful as a knowledge and reasoning modeling tool: in particular, we demonstrate their use to support what-if reasoning capabilities, which are of primary importance in decision support. Formal backing to this approach is provided by the $AFRA$ formalism, a recently proposed extension of Dung's argumentation framework. A literature example concerning a decision problem about medical treatments is adopted to illustrate the approach.

Introduction

Providing decision support is not just a matter of identifying a decision to be suggested. Good decision support, similarly to good human advice, should involve explanation and interaction with decision makers (Girle et al. 2003):

1. the advice should be presented in a form which can be readily understood by decision makers;
2. there should be ready access to both information and reasoning underpinning the advice;
3. if decision support involves details which are unusual to the decision maker, it is of primary importance that s/he can discuss these details with his advisor.

In particular, the second point requires transparency of the reasoning leading to the proposed advice about what to do. Reasoning about what to do is often called "practical reasoning", an important investigation subject in the "Argumentation in AI" research community. In this context, two main questions can be identified: on the one hand, appropriate schemes for the representation of knowledge and reasoning patterns have to be defined, on the other hand mechanisms to compute reasoning outcomes have to be identified.

Concerning the former question (representation), the influential work of (Walton 1996) introduces the concept of "argument scheme" intended as the statement of a presumption in favor of a given conclusion, or goal. Whether this presumption stands or falls depends on the positive or negative answers to a set of "critical questions" associated with the scheme. This approach was further developed in (Atkinson, Bench-Capon, and McBurney 2006) where a refined argument scheme for practical reasoning has been proposed, encompassing the distinction between goals, which are the desired effects of an action, and values, which represent the actual underlying reasons for an agent to achieve a goal.

As to the latter question (computation), all the approaches based on argument schemes mentioned above seem to assume the existence of different reasoning levels; for example in (Atkinson, Bench-Capon, and McBurney 2006) the levels of decision outcomes, goals, and values. In this perspective the Value-based Argumentation Framework (VAF) (Bench-Capon 2003) extends Dung's argumentation framework (AF) (Dung 1995) by introducing meta-level reasoning about values. Another extension of Dung's AF, called Extended Argumentation Framework (EAF), proposed in (Modgil 2007; 2009), advocates the existence of attacks to attacks, rather than just to arguments. Relationships between these approaches are discussed in (Bench-Capon and Modgil 2008) where it is shown that any instance of VAF can be put in correspondence with an instance of EAF. Recently a formalism called Argumentation Framework with Recursive Attacks ($AFRA$) has been introduced (Baroni et al. 2009), where a more general notion of attack to attack than in EAF is considered. In fact, $AFRA$ encompasses attacks to attacks recursively without constraints, while in EAF attacks to attacks can not be attacked in turn.

In this paper we propose a model of decision support in the context of an argumentation-based approach. This allows to carry out the task of computing the decision(s) to be proposed relying on a sound argumentation framework, thus achieving transparency of the computation process and the capability of dealing with what-if reasoning. In order to achieve this goal, we proceed in two steps: first we introduce an articulated set of concepts for representing a Decision Support Problem; then, we focus on the computation of the decision to be suggested, relying on the AFRA formalism (Baroni et al. 2009). An example firstly proposed in (Atkinson, Bench-Capon, and McBurney 2006) is used throughout the paper to illustrate the proposed approach.

An Example

We will illustrate our proposal by referring to an example on the treatment of heart disease first introduced in (Modgil and Fox 2004) and then considered also in (Modgil 2006) and (Atkinson, Bench-Capon, and McBurney 2006). We adopt a slightly modified version of the formulation given in (Atkinson, Bench-Capon, and McBurney 2006).

The action to be chosen concerns the treatment for a patient threatened by blood clotting: the relevant goal is obtaining a low platelet adhesion. It is assumed that in the available knowledge base there are only two actions to achieve that goal: "administer aspirin" (**A2**) and "administer chlopidogrel" (**A3**). Both choices achieve the goal of reducing blood clotting which will promote the value of safety (**V2**). But, obviously, there is another possible default decision, namely "do nothing" (**A1**) achieving the goal of having small expenses which will in turn promote the value of cost (**V1**). Since we assume that only one decision has to be adopted, the alternative choices are mutually incompatible. Obviously the final decision will be acceptable only if it achieves the goal of reducing blood clotting.

From patient's file we learn that he has a history of gastritis and we should not administer aspirin (**A4**) because it gives rise to risk of ulceration which will demote the value of safety. This risk would be eliminated by administering a proton pump inhibitor, which however is assumed to be unavailable. Clearly **A4** and **A2** are incompatible, moreover from the reasons underlying **A4** we have learned that **A2** demotes the value of safety. Therefore the evidence supporting **A4** also provides information against the relation between **A2** and the value of safety.

Suppose now that the assumption that no proton pump inhibitor is available reveals to be false and, therefore, it is possible to administer it (**A5**): in this case **A4** looses its support and **A2** is reinstated. Then, suppose that between aspirin and chlopidogrel a doctor prefers to administer aspirin (**P1**) because it is in stock and immediately available. Such a preference is not in contrast with **A3**: it states only that in this particular situation, if we have to choose between **A2** and **A3**, we would prefer **A2**.

Finally, we can determine the ultimate decision outcome by considering values. In order to achieve the goal of reducing blood clotting, which promotes the value of safety, the action of administering aspirin has to prevail on the action "do nothing". To this purpose we have to consider the value of safety in a strong form stating that "we must promote the value of safety". Then, the final decision outcome, according to (Atkinson, Bench-Capon, and McBurney 2006) will be $\{\mathbf{A5}, \mathbf{A2}\}$, namely we should administer aspirin and the proton pump inhibitor.

Representing a decision support problem

A decision support problem may be formalized adopting an argument-based approach where two basic notions, namely arguments and attacks, are encompassed. For a suitable representation, both notions need to be specialized: arguments of different sorts can be identified in relation with different reasoning levels (e.g. about goals rather than about values). This involves in turn different kinds of attack relations (possibly encompassing attacks to attacks, as it will be discussed later). Accordingly, the modeling approach we propose is based on an articulated set of concepts:

- the notion of practical argument scheme **PAS**, derived from (Atkinson, Bench-Capon, and McBurney 2006);
- the concept of practical attack scheme **PAtS**, that defines the conditions for an attack relation to hold between two instances of **PAS**;
- the concept of factual argument scheme **FAS**, defining the circumstances holding in a certain step of the reasoning process, and the related concept of factual attack scheme **FAtS** between an instance of **FAS** and an instance of **PAS**;
- the concept of value argument scheme **VAS**, asserting that a given value is in force, and the related concepts of: (1) value attack scheme **VAtS**, involving incompatible values, (2) value defence scheme **VDeS**, involving attacks from a value scheme to other attacks, and (3) value-argument attack scheme **VAAtS** involving argument schemes related through a value defence scheme;
- a preference argument scheme **PRAS**, to define a preference ordering between instances of other schemes, and the related preference attack scheme **PRAtS**, covering the cases where a preference undermines an attack;
- a must argument scheme **MAS**, which supports what-if reasoning by stating a value to be promoted above all others, and the related must attack schemes **MAtS**.

We will proceed in our presentation according to the above plan: each scheme will be introduced as a tuple of entities, whose meaning will be commented case by case.

First, the following definition provides a modified version of the Practical Argument Scheme proposed in (Atkinson, Bench-Capon, and McBurney 2006) (in particular, we omit the future circumstances which are caused by an action and consider only the goals it achieves).

PAS: `circumstance`: C,
`action`: A,
`goal`: G,
`value`: V,
`sign`: +/-.

The scheme means that "in the circumstances C, the suggested action is A, which achieves the goal G, which, depending on sign, promotes or demotes the value V". We assume that the suggested action may be also a "negative" action $\neg$ `A`, with the meaning "it is suggested not to perform action `A`". We assume also that $\neg\neg$`A` = `A`.

We can then introduce two attack schemes between instances of **PAS**. Any attack scheme has a source, namely an instance of a scheme with the role of attacker, and a target, namely an instance of a scheme which is attacked, and specifies the conditions under which the attack takes place.

PAtS1: `source`: an instance of **PAS**,
`target`: an instance of **PAS**,
`conditions`: `source.action` $\neq$ `target.action` and both `source.action` and `target.action` are positive.

PAtS2: `source`: an instance of **PAS**,
`target`: an instance of **PAS**,
`conditions`: `source.action` = ¬ `target.action`.

PAtS1 corresponds to the case where distinct positive actions are incompatible and therefore we have to choose exactly one action. **PAtS2** corresponds to the case where the source **PAS** suggests not to perform the action supported by the target **PAS** and vice versa.

We now introduce a simple factual argument scheme, corresponding to the assertion that the circumstances `C` hold. We assume that it is possible to negate that certain circumstances hold, using again ¬ as negation symbol.

FAS: `circumstances`: C.

We can then introduce a factual attack scheme: a factual assertion attacks a **PAS** by negating its circumstances.

FAtS: `source`: an instance of **FAS**,
`target`: an instance of **PAS**,
`conditions`: `source.circumstances` = ¬ `target.circumstances`.

We introduce also a simple value argument scheme, representing the fact that the value `V` is in force.

VAS: `value`: V.

Several attack schemes involve instances of **VAS**. First, we assume that a symmetric incompatibility relation, denoted as I, is defined among values: incompatible values attack each other, giving rise to a value attack scheme.

VAtS: `source`: an instance of **VAS**,
`target`: an instance of **VAS**,
`conditions`: (`source.value`, `target.value`) $\in I$.

Then we consider the defence of practical arguments based on values, following an idea proposed in (Modgil 2007; Bench-Capon and Modgil 2008). In words, a **VAS** argument, let say **V1**, defends an instance of **PAS**, let say **Pa1**, against an instance of attack **PAtS1** or **PAtS2** whose target is **Pa1**. The defence takes place if the value of **V1** coincides with that of **Pa1** and is different from the value of the source of the attack against **Pa1**. This is represented by the following scheme.

VDeS1: `source`: an instance of **VAS**,
`target`: an instance of **PAtS1** or **PAtS2**,
`conditions`: `target.target.value` = `source.value` and `target.source.value` ≠ `source.value`.

As values may defend arguments, they may analogously defend attacks. In fact, if the source of an attack is a practical argument, then it promotes or demotes a particular value. Such an attack is again strictly related to such value, therefore the value must defend this attack against the attacks it may receive from other values. To exemplify, an instance of **VDeS1**, let say **Vd1**, may be attacked by an instance of **VAS**, let say **V2**, if the value of **V2** is different from that of the source of **Vd1** and coincides with the value of the source of the instance of **PAtS1** or **PAtS2** which is the target of the attack **Vd1**. This is expressed by the following scheme.

VDeS2: `source`: an instance of **VAS**,
`target`: an instance of **VDeS1**,
`conditions`: `target.source` ≠ `source` and `target.target.source.value` = `source.value`.

Since both **VDeS1** and **VDeS2** schemes represent a defence originated by a value in favor of a practical argument, we can see them as specializations of an abstract defence scheme that we call **VDefence**.

VDefence: `defending`: an instance of **VAS**,
`defended`: an instance of **PAS**.

In particular, letting **X** be an instance of **VDeS1**, we have **X**.`defending` = **X**.`source` and **X**.`defended` = **X**.`target.target`. On the other hand, letting **Y** be an instance of **VDeS2** we have: **Y**.`defending` = **Y**.`source` and **Y**.`defended` = **Y**.`target.target.source`.

We can now introduce the last attack scheme concerning practical arguments which is strictly related with an issue pointed out in (Atkinson, Bench-Capon, and McBurney 2006). This concerns the case where a practical argument suggests not to perform an action `A`, since it demotes a value.

VAAtS: `source`: an instance of **PAS**,
`target`: an instance of **VDefence**,
`conditions`: `source.circumstance` = `target.defended.circumstance` and `source.action` = ¬ `target.defended.action` and `source.goal` = `target.defended.goal` and `source.value` = `target.defended.value` and `source.sign` = − and `target.defended.sign` = +.

This scheme applies in cases where an instance **Pa1** of **PAS** (the source) has a strict relation with an instance **Pa2** of **PAS** defended by an instance of **VDefence** (**Pa2** corresponds to `target.defended`). **Pa1** and **Pa2** share the same circumstances and the same goal, they are related to the same value but with different signs (**Pa1** demotes it, while **Pa2** promotes it), and **Pa2** suggests not to execute the action supported by **Pa1**. In a word, **Pa2** tells that in order to promote the value of **Pa1** one has actually not to perform the action suggested by **Pa1**, i.e. under the current circumstances **Pa1** would demote, instead of promoting, its value. Consequently, any instance of **VDefence**, which defends **Pa2** on the basis of the value it promotes, is attacked in turn by **Pa1** (note that, given the above assumptions, it also holds that **Pa1** attacks **Pa2** according to the **PAtS2** scheme).

Let us turn now to the representation of preferences. A preference argument scheme simply corresponds to stating that an argument is preferred to another one.

PRAS: `preferred`: P,
`notpreferred`: nP.

Following (Modgil 2007), we can then define an attack scheme based on preferences: an instance of **PRAS**, let say

Pref1, attacks an attack, let say **Att1**, if **Pref1** states that the target of **Att1** is preferred to the source of **Att1**:

PRAtS: `source`: an instance of **PRAS**,
`target`: an instance of **PAtS1**, or **PAtS2**, or **VAtS**,
`conditions`: `source.preferred` = `target.target`
and `source.notpreferred` = `target.source`.

In order to formalize what-if reasoning situations, we define a simple "must" argument scheme concerning a single value which must be promoted over all others.

MAS: `value`: V.

A **MAS** argument supports what-if reasoning since it gives rise to a scenario where we can compute the consequences of adopting a value as main reference. Different scenarios can be considered by assuming alternative **MAS** arguments: only one **MAS** argument can be assumed at a time. This is a restrictive, though potentially useful, assumption, that we plan to relax in the future, considering sets of values to be promoted and preferences among them.

In relation with **MAS** we introduce two attack schemes.

MAtS1: `source`: an instance of **MAS**,
`target`: an instance of **VAtS**,
`conditions`: `source.value` = `target.target.value`
and `source.value` ≠ `target.source.value`.

MAtS2: `source`: an instance of **MAS**,
`target`: an instance of **VDeS2**,
`conditions`: `source.value` = `target.target.source.value`
and `source.value` ≠ `target.source.value`.

In both schemes an instance of **MAS**, call it **M1**, defends an instance of **VAS** with the same value. In particular, in the **MAtS1** scheme, **M1** attacks an instance of **VAtS** attack (let say **Va1**) such that the target of **Va1** has the same value as **M1** and the value of the source of **Va1** is different from the one of **M1**. In the **MAtS2** scheme, **M1** attacks an instance of **VDeS2** attack (let say **Vd2**) since **Vd2** attacks another attack whose source is based on the same value as **M1**, while the source of **Vd2** is a value different from the one of **M1**.

Summing up, we may define an Argumentation Framework representing a Decision Support Problem as a tuple including instances of all schemes introduced above.

Definition 1 ($AFDSP$). *An Argumentation Framework for Decision Support Problem ($AFDSP$) is a 10-ple $\langle \mathcal{A}_{PAS}$, $\mathcal{A}_{PRAS}$, $\mathcal{A}_{VAS}$, $\mathcal{A}_{FAS}$, $\mathcal{A}_{MAS}$, $\mathcal{R}_{PAS}$, $\mathcal{R}_{PRAS}$, $\mathcal{R}_{VAS}$, $\mathcal{R}_{FAS}$, $\mathcal{R}_{MAS}\rangle$ s.t.:*

- *$\mathcal{A}_{PAS}$ is a set of instances of **PAS**;*
- *$\mathcal{A}_{PRAS}$ is a set of instances of **PRAS**;*
- *$\mathcal{A}_{VAS}$ is a set of instances of **VAS**;*
- *$\mathcal{A}_{FAS}$ is a set of instances of **FAS**;*
- *$\mathcal{A}_{MAS}$ is a set of instances of **MAS**;*
- *$\mathcal{R}_{PAS}$ is a set of instances of **PAtS1** and **PAtS2**;*
- *$\mathcal{R}_{PRAS}$ is a set of instances of **PRAtS**;*
- *$\mathcal{R}_{VAS}$ is a set of instances of **VAtS**, **VDeS1**, **VDeS2**, **VAAtS**;*
- *$\mathcal{R}_{FAS}$ is a set of instances of **FAtS**;*
- *$\mathcal{R}_{MAS}$ is a set of instances of **MAtS1**, and **MAtS2**.*

We turn now to the problem of providing a formal backing to the proposed representation in order to support the computation of the decision to be suggested.

Computing the outcome of the decision process

According to our approach, once a Decision Support Problem has been represented using the concepts defined in the previous section, we can compute the relevant decision outcomes relying on the formal notion of extensions of an argumentation framework. To this purpose, we rely on a new formalism called $AFRA$ (Baroni et al. 2009) which extends Dung's AF by allowing attacks to attacks in a recursive way. We briefly recall in the following the main notions of AFRA.

Definition 2 ($AFRA$). *An Argumentation Framework with Recursive Attacks ($AFRA$) is a pair $\langle \mathcal{A}, \mathcal{R}\rangle$ where $\mathcal{A}$ is a set of arguments and $\mathcal{R}$ is a set of attacks, namely pairs $(A, \mathscr{X})$ s.t. $A \in \mathcal{A}$ and ($\mathscr{X} \in \mathcal{R}$ or $\mathscr{X} \in \mathcal{A}$).*
Given an attack $\alpha = (A, \mathscr{X}) \in \mathcal{R}$, we will say that A is the source of α, denoted as $src(\alpha) = A$ and $\mathscr{X}$ is the target of α, denoted as $trg(\alpha) = \mathscr{X}$.

We start substantiating the role played by attacks by introducing a notion of defeat which regards attacks, rather than their source arguments, as the subjects able to defeat arguments or other attacks, as encompassed by Definition 3.

Definition 3 (Direct Defeat). *Let $\langle \mathcal{A}, \mathcal{R}\rangle$ be an $AFRA$, $\mathscr{V} \in \mathcal{R}$, $\mathscr{W} \in \mathcal{A} \cup \mathcal{R}$, then $\mathscr{V}$ directly defeats $\mathscr{W}$ iff $\mathscr{W} = trg(\mathscr{V})$.*

Moreover, since we are interested also in how attacks are affected by other attacks, we introduce a notion of indirect defeat for an attack, corresponding to the situation where its source receives a direct defeat.

Definition 4 (Indirect Defeat). *Let $\langle \mathcal{A}, \mathcal{R}\rangle$ be an $AFRA$, $\mathscr{V} \in \mathcal{R}$, $\mathscr{W} \in \mathcal{A}$, if $\mathscr{V}$ directly defeats $\mathscr{W}$ then $\forall \alpha \in \mathcal{R}$ s.t. $src(\alpha) = \mathscr{W}$, $\mathscr{V}$ indirectly defeats α.*

A defeat is a direct or indirect defeat.

Definition 5 (Defeat). *Let $\langle \mathcal{A}, \mathcal{R}\rangle$ be an $AFRA$, $\mathscr{V} \in \mathcal{R}$, $\mathscr{W} \in \mathcal{A} \cup \mathcal{R}$, then $\mathscr{V}$ defeats $\mathscr{W}$, denoted as $\mathscr{V} \rightarrow_R \mathscr{W}$, iff $\mathscr{V}$ directly or indirectly defeats $\mathscr{W}$.*

The definition of conflict-free set follows directly.

Definition 6 (Conflict–free). *Let $\langle \mathcal{A}, \mathcal{R}\rangle$ be an $AFRA$, $\mathcal{S} \subseteq \mathcal{A} \cup \mathcal{R}$ is conflict–free iff $\nexists \mathscr{V}, \mathscr{W} \in \mathcal{S}$ s.t. $\mathscr{V} \rightarrow_R \mathscr{W}$.*

The definition of acceptability is similar to the traditional one, but involves both arguments and attacks.

Definition 7 (Acceptability). *Let $\langle \mathcal{A}, \mathcal{R}\rangle$ be an $AFRA$, $\mathcal{S} \subseteq \mathcal{A} \cup \mathcal{R}$, $\mathscr{W} \in \mathcal{A} \cup \mathcal{R}$, $\mathscr{W}$ is acceptable w.r.t. $\mathcal{S}$ iff $\forall \mathscr{Z} \in \mathcal{R}$ s.t. $\mathscr{Z} \rightarrow_R \mathscr{W}$ $\exists \mathscr{V} \in \mathcal{S}$ s.t. $\mathscr{V} \rightarrow_R \mathscr{Z}$.*

On this basis, also the definitions of *admissible set* and *preferred extension* are analogous to the traditional ones.

Definition 8 (Admissible set - Preferred Extension). *Let $\langle \mathcal{A}, \mathcal{R} \rangle$ be an $AFRA$, $\mathcal{S} \subseteq \mathcal{A} \cup \mathcal{R}$ is admissible iff it is conflict–free and each element of $\mathcal{S}$ is acceptable w.r.t. $\mathcal{S}$. A preferred extension is a maximal (w.r.t. set inclusion) admissible set.*

We propose a natural correspondence from an $AFDSP$ to an $AFRA$: the instances of argument schemes in $AFDSP$ compose the set of arguments in $AFRA$ and the instances of attack schemes in $AFDSP$ give rise to the attack relation in $AFRA$. Formally, letting $\Phi = \langle \mathcal{A}_{PAS}, \mathcal{A}_{PRAS}, \mathcal{A}_{VAS}, \mathcal{A}_{FAS}, \mathcal{A}_{MAS}, \mathcal{R}_{PAS}, \mathcal{R}_{PRAS}, \mathcal{R}_{VAS}, \mathcal{R}_{FAS}, \mathcal{R}_{MAS} \rangle$ be an $AFDSP$, the corresponding $AFRA$ is defined as $\Gamma = \langle \mathcal{A}, \mathcal{R} \rangle$ s.t. $\mathcal{A} = \mathcal{A}_{PAS} \cup \mathcal{A}_{PRAS} \cup \mathcal{A}_{VAS} \cup \mathcal{A}_{FAS} \cup \mathcal{A}_{MAS}$; and $\mathcal{R} = \mathcal{R}_{PAS} \cup \mathcal{R}_{PRAS} \cup \mathcal{R}_{VAS} \cup \mathcal{R}_{FAS} \cup \mathcal{R}_{MAS}$.

In order to keep a correspondence between our approach and the one by (Atkinson, Bench-Capon, and McBurney 2006), we exploit the notion of preferred extension to define the outcomes of the decision process. In general, adopting a multiple status semantics is compatible with the idea that several alternative courses of actions may be considered and that the decision maker is encouraged to evaluate and criticize the advice provided by the system. More precisely, every preferred extension of an $AFRA$ is a set of arguments and attacks which can be regarded altogether as a reasonable and defendable position. The instance(s) of practical arguments included in a preferred extension correspond to the suggested action(s). In general, several distinct preferred extensions may exist, corresponding to alternative and equally defendable courses of actions. Assuming different **MAS** arguments, different results may be obtained, thus supporting what-if reasoning.

The example revisited

In this section we apply our approach to the medical example previously presented by defining an $AFDSP$ Φ for its representation. First of all, we have to formalize practical arguments through instances of **PAS**.

A1: `circumstance`: given patient's situation,
`action`: we should do nothing,
`goal`: having small expense,
`value`: cost,
`sign`: +.

A2: `circumstance`: given patient's situation,
`action`: we should administer aspirin,
`goal`: reducing blood clotting,
`value`: safety,
`sign`: +.

A3: `circumstance`: given patient's situation,
`action`: we should administer chlopidogrel,
`goal`: reducing blood clotting,
`value`: safety,
`sign`: +.

A4: `circumstance`: proton pump unavailable,
`action`: we should not administer aspirin,
`goal`: risk of ulceration,
`value`: safety,
`sign`: −.

The example also includes an instance of **FAS** and **PRAS**.

A5: `circumstances`: a proton pump is available.

P1: `preferred`: **A2**,
`notpreferred`: **A3**.

The considered values give rise to instances of **VAS**.

V1: `value`: cost.

V2: `value`: safety.

Summing up, in Φ we have: $\mathcal{A}_{PAS} = \{\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{A4}\}$, $\mathcal{A}_{PRAS} = \{\mathbf{P1}\}$, $\mathcal{A}_{FAS} = \{\mathbf{A5}\}$, and $\mathcal{A}_{VAS} = \{\mathbf{V1}, \mathbf{V2}\}$.

Let us examine now the attack relations. We assume that only one action among those supported by **A1**, **A2**, and **A3** can be adopted. Therefore **A1**, **A2**, and **A3** attack each other according to the **PAtS1** scheme. Moreover **A4** supports the negation of the action supported by **A2**. Hence **A2** and **A4** attack each other according to the **PAtS2** scheme. **A5** negates the circumstance of **A4**, thus attacking it according to the **FAtS** scheme. Moreover, given the preference for **A2** over **A3**, **P1** attacks the attack from **A2** to **A3** according to the **PRAtS** scheme. Summing up, we have: $\mathcal{R}_{PAS} = \{(\mathbf{A1}, \mathbf{A2}), (\mathbf{A2}, \mathbf{A1}), (\mathbf{A3}, \mathbf{A1}), (\mathbf{A1}, \mathbf{A3}), (\mathbf{A2}, \mathbf{A3}), (\mathbf{A3}, \mathbf{A2}), (\mathbf{A4}, \mathbf{A2}), (\mathbf{A2}, \mathbf{A4})\}$; $\mathcal{R}_{FAS} = \{(\mathbf{A5}, \mathbf{A4})\}$; $\mathcal{R}_{PRAS} = \{(\mathbf{P1}, (\mathbf{A3}, \mathbf{A2}))\}$.

Let us turn to attacks involving values (note that **V1** and **V2** are assumed not to be incompatible *per se*, hence there are no instances of the **VAtS** scheme). According to the **VDeS1** scheme, **V1** attacks the attacks directed to **A1** whose source promotes a different values, and so does **V2** with respect to attacks against **A2** and **A3**. This gives rise to the following set of attacks: $\mathcal{R}^1_{VAS} = \{\mathbf{V1}, (\mathbf{A3}, \mathbf{A1})), (\mathbf{V1}, (\mathbf{A2}, \mathbf{A1})), (\mathbf{V2}, (\mathbf{A1}, \mathbf{A2})), (\mathbf{V2}, (\mathbf{A1}, \mathbf{A3}))\}$.

In turn, according to **VDeS2**, **V1** attacks the instances of **VDeS1** based on a different value (actually **V2**) and whose target is an attack whose source promotes **V1** and a dual consideration applies to **V2**, giving rise to the following set of attacks: $\mathcal{R}^2_{VAS} = \{(\mathbf{V1}, (\mathbf{V2}, (\mathbf{A1}, \mathbf{A2}))), (\mathbf{V1}, (\mathbf{V2}, (\mathbf{A1}, \mathbf{A3}))), (\mathbf{V2}, (\mathbf{V1}, (\mathbf{A3}, \mathbf{A1}))), (\mathbf{V2}, (\mathbf{V1}, (\mathbf{A2}, \mathbf{A1})))\}$. Furthermore, according to **VAAtS**, **A4** attacks the defences of **A2** based on safety giving rise to $\mathcal{R}^3_{VAS} = \{(\mathbf{A4}, (\mathbf{V2}, (\mathbf{A1}, \mathbf{A2}))), (\mathbf{A4}, (\mathbf{V2}, (\mathbf{V1}, (\mathbf{A2}, \mathbf{A1}))))\}$.

In summary, $\mathcal{R}_{VAS} = \mathcal{R}^1_{VAS} \cup \mathcal{R}^2_{VAS} \cup \mathcal{R}^3_{VAS}$.

Finally, we assume that a **MAS** argument concerning safety is assumed.

MUST V2: `value`: safety.

Then $\mathcal{A}_{MAS} = \{\textbf{MUST V2}\}$ and, according to the **MAtS2** scheme two further attacks arise $\mathcal{R}_{MAS} = \{\textbf{MUST V2}, (\mathbf{V1}, (\mathbf{V2}, (\mathbf{A1}, \mathbf{A2})))), (\textbf{MUST V2}, (\mathbf{V1}, (\mathbf{V2}, (\mathbf{A1}, \mathbf{A3}))))\}$.

The resulting Φ is shown in Fig. 1, where arrows represent instances of **PAtS**, **FAtS**, **PRAtS**, and **VAAtS**; dotted arrows represent instances of **VAtS** and **VDeS**; and dotted-dashed arrows represent instances of **MAtS**.

Given Φ, we directly obtain its corresponding $AFRA$ Γ. The arguments included in its (unique) preferred extension are: $\{\textbf{MUST V2}, \mathbf{V2}, \mathbf{V1}, \mathbf{P1}, \mathbf{A5}, \mathbf{A2}\}$. This corresponds to suggesting the actions of administering aspirin and proton pump inhibitor, as expected. We omit the relevant formal derivation due to space limitations. Intuitively the reader

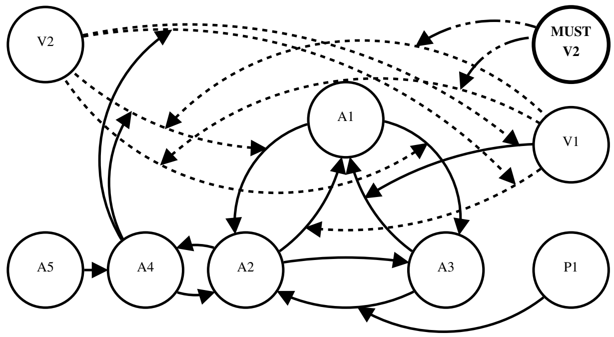

Figure 1: A graphical representation of Φ

may consider the simplified representation provided in Figure 2, where attacks which do not survive the attacks they receive are suppressed and the arguments included in the preferred extension are shown in grey. As a final remark concerning what-if reasoning, note that if we would have assumed a different **MAS** argument concerning cost, we would have obtained a different outcome, with **A1** accepted and the "do nothing" action supported.

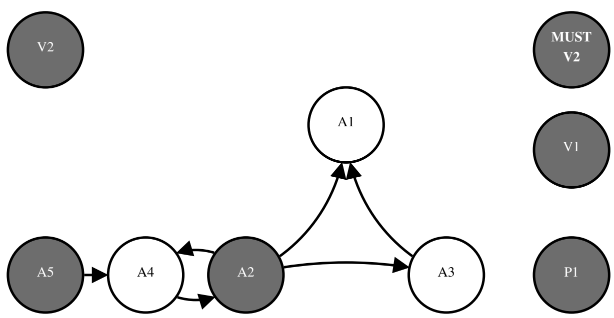

Figure 2: The only preferred extension of Γ

Discussion and conclusions

In this paper we have illustrated a proposal concerning the representation of decision support problems through an argument-based approach. This work is at an early stage of development and the proposed argument and attack schemes are far from being unquestionable and complete. In spite of these limitations, the paper provides two contributions:

- it shows the role that may be played by attack schemes, an issue which, as to our knowledge, has received so far only limited attention in the literature;
- it demonstrates that different levels of attacks to attacks are useful to capture some intuitive patterns of practical reasoning and to support what-if capabilities, thus emphasizing the role of the $AFRA$ formalism.

As to related works, the example used in the paper was first introduced in (Modgil and Fox 2004) in the context of a medical multi-agent systems where agents debate about the therapy to be administered. Accordingly, this work is more focused on issues concerning agent dialogues than on modeling decision support. Using the same example, the work in (Atkinson, Bench-Capon, and McBurney 2006) is focused on an approach to practical reasoning, regarded as presumptive justification of a course of action, depending on the answers to a set of critical questions. An interaction protocol (PARMA) is also developed for agent dialogue about the proposed action. This approach adopts an argument scheme, along with relevant critical questions, where relations among practical arguments, goals and values are represented in a structured way. Encompassing attacks to attacks is however not considered in this work, based on the Value-based Argumentation Framework. Neither (Atkinson, Bench-Capon, and McBurney 2006) nor (Modgil and Fox 2004) address the issue of what-if reasoning. Due to space limitations, we have not discussed in this paper the relationship between attack schemes and critical questions, an important issue which deserves further analysis. Other future research directions include an extensive investigation of argument and attack schemes for decision support, accompanied by theoretical developments of the $AFRA$ formalism.

References

Atkinson, K.; Bench-Capon, T. J. M.; and McBurney, P. 2006. Computational representation of practical argument. *Synthese* 152(2):157–206.

Baroni, P.; Cerutti, F.; Giacomin, M.; and Guida, G. 2009. Encompassing attacks to attacks in abstract argumentation frameworks. In *Proc. of ECSQARU 2009*, 83–94.

Bench-Capon, T. J. M., and Modgil, S. 2008. Integrating object and meta-level value based argumentation. In *Proc. of COMMA 2008*, 240–251.

Bench-Capon, T. J. M. 2003. Persuasion in practical argument using value based argumentation frameworks. *J. of Logic and Computation* 13(3):429–448.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *AIJ* 77(2):321–357.

Girle, R.; Hitchcock, D. L.; McBurney, P.; and Verheij, B. 2003. Decision support for practical reasoning: A theoretical and computational perspective. In Reed, C., and Norman, T. J., eds., *Argumentation Machines. New Frontiers in Argument and Computation*, 55–84. Kluwer.

Modgil, S., and Fox, J. 2004. A guardian agent approach to safety in medical multi-agent systems. In *Proc. of SEAMAS'04 at AAMAS 2004*.

Modgil, S. 2006. Value based argumentation in hierarchical argumentation frameworks. In *Proc. of COMMA 2006*, 297–308.

Modgil, S. 2007. An abstract theory of argumentation that accommodates defeasible reasoning about preferences. In *Proc. of ECSQARU 2007*, 648–659.

Modgil, S. 2009. Reasoning about preferences in argumentation frameworks. *AIJ* 173(9-10):901–934.

Walton, D. N. 1996. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates.