Robust Semantics for Argumentation Frameworks

H. Jakobovits and D. Vermeir
Free University of Brussels, VUB
Dept. of Computer Science
Pleinlaan 2, Brussels 1050, Belgium

Abstract

We suggest a so-called "robust" semantics for a model of argumentation which represents arguments and their interactions, called "argumentation frameworks". We study a variety of additional definitions of acceptability of arguments; we explore the properties of these definitions; we describe their inter-relationships: e.g. robust models can be characterized using the minimal (well-founded) models of a meta-framework. The various definitions of acceptability of argument sets can all deal with contradiction within an argumentation framework.

Keywords: Argumentation framework, semantics

1 Introduction

In this paper we present semantics for a formal model of argumentation. As in other works such as [Pol94] and [Dun95], we abstract from the actual contents and form of the arguments themselves, and rather concentrate on the analysis of *interactions between arguments*.

Argumentation-theoretic interpretations and proof-procedures are applicable in practical reasoning, legal reasoning ([KT96],[PS95]), mediation systems ([GK96],[BG94]), decision-making systems ([KPG96]), and are especially useful for reasoning among electronic agents and in deductive databases. Moreover, as pointed out in [BDKT97], this approach unifies and generalizes many existing approaches to defeasible reasoning, including several systems of non-monotonic logic, default logic, and logic programming. For example, the approach presented in this paper generates, by a simple and natural mapping (see [JV96]), a new semantics for logic programs.

Within a given framework of interacting arguments, there might be one or several sets of conclusions that are deemed acceptable. The selected set(s) must satisfy certain criteria, such as consistency, coherence, etc. These criteria have been successfully formalized in a rigorous manner in [Dun95] and [BDKT97], and were extended to include defeasible interactions, in [Pra96]. The theory and its extension are based on the idea that a statement (argument) is admissible if it can be argued successfully against contesting arguments. However, as [Dun95] and [BDKT97] point out, the theory lacks facilities to satisfactorily deal with arguments that, directly or indirectly, contradict

themselves. This problem is approached in [Pol94], where it is suggested to take into account only a subset of the given arguments, which entail no contradiction.

In this paper, we present an alternative theory of acceptability in argumentation frameworks, which proposes a solution to the above problem. We show that our approach – which is based on one single simple concept called a "labeling" – unifies several previous approaches (such as preferred and stable sets of [Dun95]). We then proceed to refine this concept into various semantics which are more adapted to specific types of reasoning, and therefore apt to capturing the subtleties present in different types of problems.

The paper is organized as follows. In section 2, we define argumentation frameworks and describe some well-known semantics. An example application illustrating their weaknesses is also presented.

Section 3 contains the basic definition of our theory of argumentation, that of a "labeling", and its associated notion of acceptability for argument sets. We show, in section 3.1, how labelings can be used to define global as well as local acceptability, thus unifying seemingly unrelated traditions of looking at argumentation frameworks. In section 3.2 we motivate our notion of acceptability by showing how it can be derived from a generalization of the concept of "defendability", as defined in [Dun95].

Before presenting our main semantics, which we call the "robust semantics", in section 4.2, we refine our semantics in section 4 into various other "sub-semantics", each suited to different types of reasoning. These refinements include so-called "minimal" (sceptical) semantics, introduced in section 4.1. The robust semantics (section 4.2) is motivated by a stability criterion presented in section 4.2.1.

Robust sets include the minimal set as well as other more "credulous" sets. In section 4.2.2 we show how to associate to an argumentation framework its so-called *meta*-argumentation framework in which meta-arguments represent labelings of the original framework. It turns out that the minimal semantics of the meta-framework characterizes the robust sets of the original framework, thus providing a simple procedure to compute robust sets.

In section 5 we compare the semantics introduced in this paper with previous proposals, most of which have a natural formulation in terms of labelings. Section 6 contains a discussion of three related approaches, those of [BDKT97], [Pol94] and [KMD94], and their relationship with our semantics. Finally, in section 7, we present conclusions and directions for further research.

All the proofs of the theorems in this paper are included in the appendix.

Contents

1 Introduction **1**

2 Argumentation frameworks **3**

3 Labelings and the acceptable semantics **6**

3.1 Global vs. local acceptability 9

3.2 Defence in the acceptable semantics 11

3.3 Self-defeating arguments 14

4 Refinements of the acceptable semantics 17

4.1 The minimal semantics 17

4.2 The robust semantics 20

4.2.1 Specification for the robust semantics 20

4.2.2 The meta-argumentation framework 22

4.2.3 Fixpoint definition of the robust semantics 24

4.2.4 The robust semantics as the minimal semantics of the meta-argumentation framework 26

5 Relationships between the various semantics 27

6 Relationships with other approaches 31

6.1 Comparison of robust semantics with [BDKT97] and [Dun95] 31

6.2 Comparison of robust semantics with acceptability of [KMD94] . . . 32

6.3 Comparison of labelings with status assignments of [Pol94] 33

7 Conclusions and directions for further research 36

8 Acknowledgments 38

9 Appendix: Proofs 38

2 Argumentation frameworks

In order to be able to analyze any given argumentation scenario, one must first be able to represent the arguments under consideration and the relationships between them.

Arguments and their interactions have been modeled in [Dun95], using the simplest of mathematical structures – the graph –. With slightly different terminology from [Dun95], the model is as follows.

Definition 1 ([Dun95]) *An* **argumentation framework** *is a pair, $AF = (A, \rightsquigarrow)$, where A is a set of arguments, and $\rightsquigarrow$ is a binary relation on A, i.e. $\rightsquigarrow \subseteq A \times A$. An argument $a \in A$ is said to* **attack** *an argument $b \in A$, denoted $a \rightsquigarrow b$, iff $(a, b) \in \rightsquigarrow$. A set S of arguments attacks another set T, denoted $S \rightsquigarrow T$, if there is an argument in S which attacks an argument in T. We write $a \not\rightsquigarrow b$ if a does not attack b. Similarly, for S and T sets of arguments, $S \not\rightsquigarrow T$ means that no element from S attacks an element in T.*

Once a model of argumentation is determined, the next objective of a theory of argumentation is to choose a "reasonable" subset of arguments from the given set. A well-known semantics for argumentation frameworks is given by the admissibility semantics, which has been suggested as follows:

Definition 2 ([Dun95]) *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. A set S of arguments is* **consistent** *iff there are no arguments $a, b \in S$ such that $a \rightsquigarrow b$. An argument a is* **defended** *by a set S of arguments iff any argument $b \in A$ which attacks a, is counterattacked by S. A consistent set S of arguments is* **admissible** *iff each argument in S is defended by S. A consistent set S of arguments is* **preferred** *iff it is maximal admissible. A consistent set S of arguments is said to be* **stable** *iff S attacks every argument that does not belong to S.*

Example 1 Consider the discussion, presented in [GK96], between a husband and wife about which car to buy. The husband, who has wanted a fast sports car ever since he finished law school, wants to buy a Porsche. The wife, whose priority is the safety of the children, wants to buy a Volvo station wagon. Schematically, the positions can be represented as arguments in an argumentation framework:

a = We should buy a Porsche.
b = We should buy a Volvo station wagon.
c = A sporty image is the most important criterion in choosing a car.
d = Safety is the most important criterion in choosing a car.

Since a Porsche is more sporty than a Volvo station wagon, argument c attacks argument b. Similarly, since a Volvo station wagon is safer than a Porsche, argument d attacks argument a. Clearly, a and b contradict each other, as do c and d. Thus, the argumentation framework generated by these positions is as shown in figure 1.

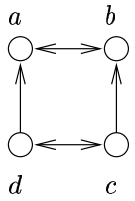


Figure 1: The argumentation framework of example 1

The admissible sets of this argumentation framework are $\{a, c\}$ and $\{b, d\}$. Each of these admissible sets represents a coherent stand.

In example 1 the admissibility semantics produces two reasonable sets of conclusions. Next we present an example which requires a more general semantics.

Application: the stable-tennis-doubles problem

Argumentation frameworks can be used to model a wide range of situations, some of which are not even linked to discussion. Here we present a variant of the stable marriage problem (see, for example, [Sed90]), which we call **the stable-tennis-doubles problem**, and which we define as follows: suppose there is a given set S of available tennis players that are candidates for being paired up into teams to play

doubles matches. Let the players express any preferences which they might have between possible partners, i.e for each distinct $player_i, player_j, player_k \in S$, $player_i$ can state that he prefers having $player_j$ to $player_k$ as a teammate. The objective of the stable-tennis-doubles problem is to arrange a set of pairs of teammates in such a way that the stated preferences are respected as much as possible. The stable-tennis-doubles problem can be described by an argumentation framework as follows: if $player_i$ prefers $player_k$ to $player_j$ then $player_i$ believes that $\{player_k, player_i\}$ is a better team than $\{player_j, player_i\}$. Therefore, $player_k$ represents a threat to the partnership $\{player_i, player_j\}$ since, from the point of view of $player_i$, the hypothetical pair $\{player_i, player_k\}$ is an *improvement* on the pair $\{player_i, player_j\}$. This can be expressed in terms of argumentation frameworks by saying that the pair $\{player_i, player_k\}$ is an attack on the pair $\{player_i, player_j\}$. [1]

A solution to the stable-tennis-doubles problem is a choice of an acceptable set(s) of so-called "stable" pairs of players, a set of rejected pairs, and perhaps a set of undecided pairs. The stable pairs are those whose improvements are all rejected, and the rejected pairs are those that have an improvement that is not rejected.

A set(s) of stable pairs is determined by a semantics of the corresponding argumentation framework $AF = (A, \rightsquigarrow)$, where A is the proposed set of pairs of players, and $\forall p, q \in T, p \rightsquigarrow q$ iff p is an improvement on q.

Example 2 Consider the following stable-tennis-doubles problem: suppose that there is a particular tennis tournament in which some of the participating players are Becker, Chang, Sampras, Agassi, Rafter and Chesnokov. Suppose that for the doubles matches, the organizational body of the tournament suggests the following set of possible pairs which involve these players:

$$\begin{aligned} a &= \{Becker, Chang\} \\ b &= \{Chang, Sampras\} \\ c &= \{Sampras, Becker\} \\ d &= \{Becker, Agassi\} \\ e &= \{Agassi, Rafter\} \\ f &= \{Rafter, Chesnokov\}. \end{aligned}$$

Suppose that the players express the following preferences for partners in doubles matches:

Chang prefers Becker to Sampras.
Sampras prefers Chang to Becker.
Becker prefers Sampras to Chang.
Becker prefers Sampras to Agassi.
Agassi prefers Becker to Rafter.
Rafter prefers Agassi to Chesnokov.

The argumentation framework associated to this stable-tennis-doubles problem is as shown in figure 2.

[1] The stable-tennis-doubles problem resembles the so-called "stable-marriage problem with gays", cited in [Dun95]. In that problem every person in S states a strictly ordered preference list containing all members of S while, in the stable-tennis-doubles problem, every player can state whichever preferences he desires and remain silent about players for which he has no preference remarks.

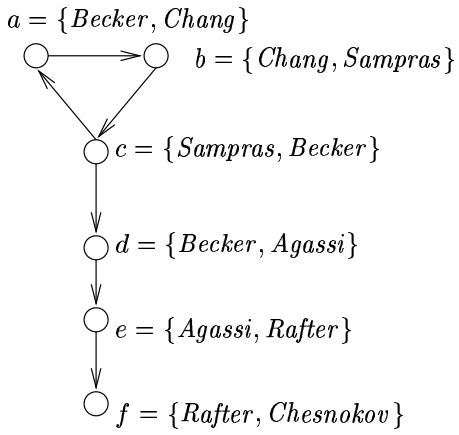


Figure 2: The stable-tennis-doubles problem of example 2

The admissible semantics of definition 2 handles the argumentation framework of figure 2 by rejecting all of the suggested pairs of players, since the only admissible set of arguments is the empty set. This happens because an admissible set must defend itself against all attacks. None of the players Chang, Sampras nor Becker form any admissible pair, since they are caught in a "preference triangle". The instability due to the preference triangle is then propagated to the the pairs involving the other players too, making all of the suggested pairs inadmissible.

In the following section we present our semantics of argumentation frameworks, which suggests additional solutions to the stable-tennis-doubles problem.

3 Labelings and the acceptable semantics

We now introduce the basic definition of our theory of argumentation, in which we associate a status to each argument in the framework. The mapping, which we call a *labeling*, is four-valued: those considered arguments that are assigned the symbol "$+$" are accepted, those that are assigned the symbol "$-$" are rejected, those that are assigned both "$+$" and "$-$" (which we denote by the symbol "$\pm$") are neither accepted nor rejected (and shall be called "undecided"), and those that are not considered at all are assigned the empty set, which signifies "don't-care".

Definition 3 *Let $(A, \rightsquigarrow)$ be an argumentation framework. A* **labeling** *is a total mapping*

$$l : A \rightarrow 2^{\{+,-\}}$$

that satisfies the following conditions:

1. $\forall a \in A$, *if* $- \in l(a)$ *then* $\exists b \rightsquigarrow a$ *such that* $+ \in l(b)$
2. $\forall a \in A$, *if* $+ \in l(a)$ *then* $\forall b \rightsquigarrow a$, $- \in l(b)$
3. $\forall a \in A$, *if* $+ \in l(a)$ *then* $\forall c$ *such that* $a \rightsquigarrow c$, $- \in l(c)$

A labeling is said to be **complete** *iff* $\forall a \in A, l(a) \neq \emptyset$. *A labeling which is not complete will sometimes be called* **partial**.

The intuition behind labelings is extremely simple: an argument that has support (i.e. contains a "+") weakens (i.e. forces a "–" on) arguments that it attacks (condition 3). Furthermore, arguments are "supported by default"; i.e. to weaken an argument, one needs support from one of its attackers (condition 1). Finally, you cannot get support for an argument unless all of its attackers have been weakened. Notice that an argument labeled $\pm$ is both supported and weakened; i.e. it is undecided. Also, whether a mapping $l : A \rightarrow 2^{\{+,-\}}$ is a labeling, is determined "locally": it suffices that each argument be labeled in a way that is consistent with (the label of) its neighbors.

These three natural rules define certain allowed symbol assignments which have the following properties: condition 1 means that if an argument is either rejected or undecided then it has an attacker which is either accepted or undecided. In the case of complete labelings, which are 3-valued, the contrapositive of this condition states that if an argument is such that everything that attacks it is rejected, then it is accepted. Condition 2 means that if an argument is either accepted or undecided then all of its attackers are either rejected or undecided. In a complete labeling, the contrapositive of this condition states that if an argument is accepted, then anything which it attacks is rejected. Condition 3 means that if an argument is either accepted or undecided then all arguments which it attacks are either rejected or undecided. In a complete labeling, the contrapositive states that if an argument is accepted, then anything that attacks it must be rejected.

With the help of the following notation and definition, we show how acceptability of argument sets is determined by labelings:

Let l be a labeling of an argumentation framework AF.

- The set $\{a \mid l(a) = +\}$ is denoted l^+.
- The set $\{a \mid l(a) = -\}$ is denoted l^-.
- The set $\{a \mid l(a) = \pm\}$ is denoted $l^{\pm}$.
- The set $\{a \mid l(a) = \emptyset\}$ is denoted $l^{\emptyset}$.

Definition 4 *Let* $AF = (A, \rightsquigarrow)$ *be an argumentation framework. A set* $S \subseteq A$ *of arguments is (partially)* **acceptable** *iff there is a labeling* l *of* AF *such that* $S = l^+$. *A set* $S \subseteq A$ *of arguments is* **completely acceptable** *iff there is a complete labeling* l *of* AF *such that* $S = l^+$. *In both of these cases we say that* l **corresponds** *to* S.

Since (completely) acceptable sets are defined solely on the basis of elementary and local rules, the resulting semantics is rather weak, in the sense that it may contain too many models. In section 4.2, where we suggest our main semantics, we shall show how unintuitive models can be filtered out.

Example 3 Consider again the stable-tennis-doubles problem presented in example 2. The labelings of the argumentation framework associated to that problem are shown in figure 3.

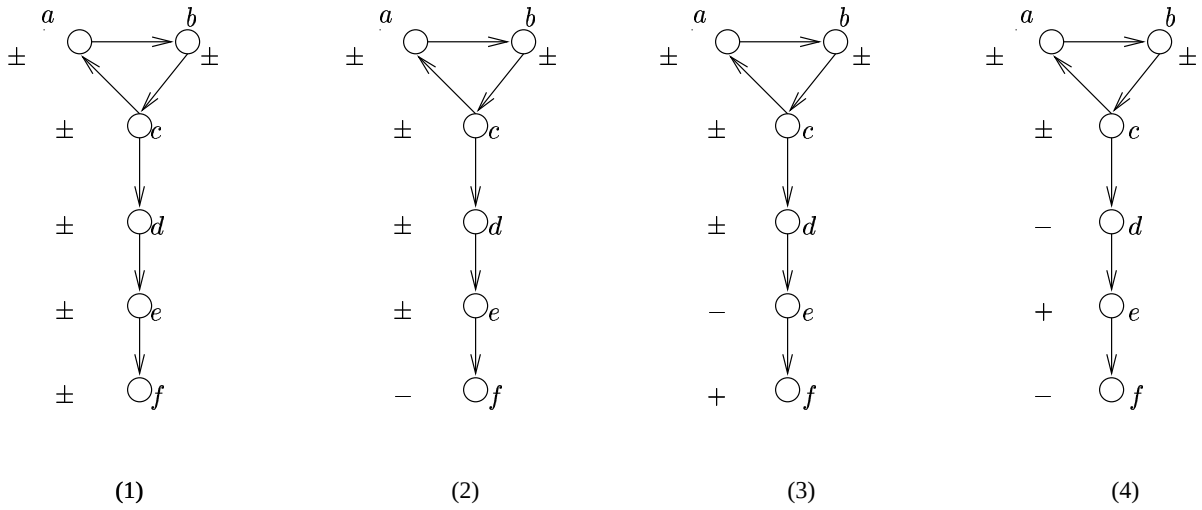


Figure 3: Complete labelings for the stable-tennis problem of example 2

Notice that all of the labelings are complete. This means that all six pairs of the framework are considered. The acceptable set corresponding to the labelings (1) and (2) is the empty set, which is the unique admissible set. The labelings (3) and (4), however, provide the acceptable sets $\{f\}$ and $\{e\}$, which are not provided by the admissible semantics.[2]

In addition to the four complete labelings shown in figure 3, the argumentation framework of example 3 also has three labelings which are not complete, as shown in figure 4.

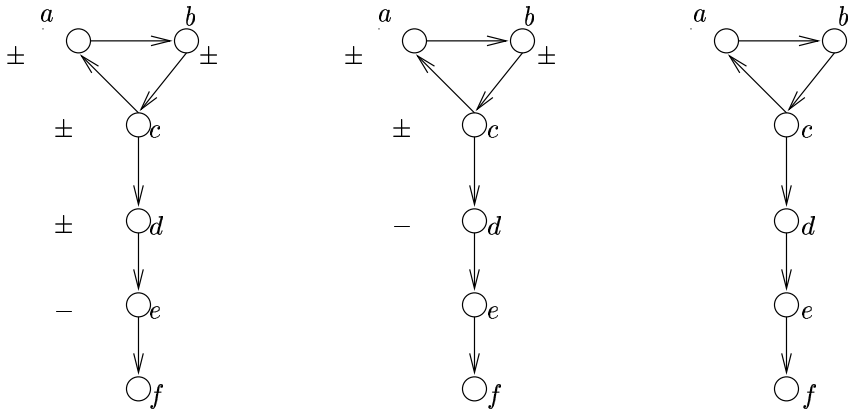


Figure 4: Partial labelings for the stable-tennis problem of example 2

All of these labelings correspond to the acceptable set $\emptyset$.

The following theorem shows that the acceptability semantics is universal.

[2]The discerning reader may notice that not all labelings in figure 3 are equally intuitive, although they all do obey the basic principles put forward in definition 3. For example, labeling (3) somehow seems less convincing than, for example, labeling (4). We will be able to pinpoint the underlying reason for the inferiority of labeling (3) in section 4.2, where robust labelings will be introduced.

Theorem 1 *Every argumentation framework has a complete labeling, and therefore a completely acceptable set.*

The next example shows how the acceptable semantics captures both sceptical and credulous reasoning.

Example 4 Consider the decision-making problem presented in [Pol94] which is as follows: suppose you have two friends, Smith and Jones, that you regard as equally reliable. Smith approaches you in the hall and says, "It is raining outside". Jones then announces, "Don't believe him. It is a fine sunny day". If you have no other evidence regarding the weather, what should you believe? It seems obvious that you should withhold belief, believing neither that it is raining nor that it is not. This is the motivation of sceptical reasoners. Credulous reasoners, on the other hand, believe that if an agent is making practical decisions, it is better to do something rather than nothing. For instance, if the agent is deciding where to have a picnic and the considerations favoring two sites are tied, it seems reasonable to choose at random.

Both of these situations can be modeled by the argumentation framework shown in figure 5.

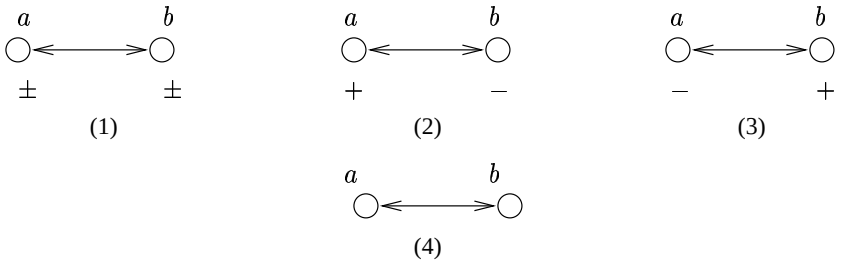


Figure 5: Sceptical and credulous labelings

The complete labeling (1) in figure 5 corresponds to the sceptical reasoner which says "I have heard the two arguments and I withhold from believing either one". The complete labeling (2), which corresponds to the completely acceptable set $\{a\}$, and the complete labeling (3), which corresponds to the completely acceptable set $\{b\}$, correspond to the credulous reasoners that choose one of the two possibilities. The labeling (4), which corresponds to the empty set, corresponds to the reasoner that doesn't care about the decision. This argumentation framework has no acceptable sets which are not completely acceptable.

3.1 Global vs. local acceptability

Complete acceptability reflects a global validity of a set of conclusions, with respect to the entire argumentation framework in question. In fact, we shall prove in theorem 6 that completely acceptable sets must include a minimal core of arguments which are beyond dispute. Partial acceptability, on the other hand, reflects a local validity of the set, and thus does not require the observer to pronounce decisions which pertain to arguments that are irrelevant, or that do not interest the observer. In this respect, acceptability is in line with the tradition of argumentation adopted in [KT96] and [KMD94]. In order for a set to be acceptable, it suffices that the set justify itself with respect to the

arguments that affect it, and that it include its ramifications. The parts of the framework which do not affect, nor are affected by, the set, need not be considered. Each of these approaches to argumentation is valid, and has its various advantages, depending on the situation. The semantics must therefore be chosen based upon the context.

Although (partial) acceptability belongs to a philosophy of argumentation is concerned with local validity of arguments, it does not allow the observer to ignore any arguments she wishes, only those that do not affect the validity of the set in consideration. It is this that distinguishes acceptable sets from, for example, subsets of completely acceptable sets. We will show in theorem 2 that if a set is acceptable then it is a subset of a completely acceptable set. This means that any locally valid set is a subset of a globally valid set, so any acceptable set can be extended to a completely acceptable set. The following example shows, however, that this property does not characterize acceptable sets; i.e. subsets of completely acceptable sets are not necessarily acceptable. The reason for this is that an acceptable set must justify its validity and must account for its consequences. In a subset of a completely acceptable set, the observer can ignore any arguments, while in an acceptable set, although one can forget about irrelevant arguments, one must care about relevant arguments.

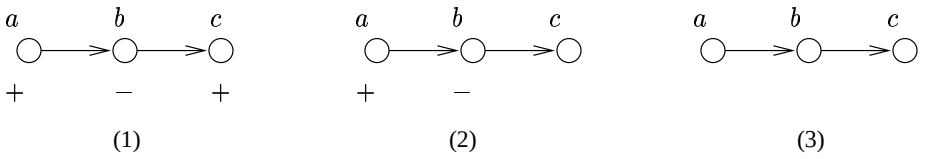


Figure 6: The argumentation framework of examples 5 and 8

Example 5 In the argumentation framework shown in figure 6, the unique completely acceptable set is $\{a, c\}$, which corresponds to the complete labeling (1). The acceptable sets are $\{a, c\}$, $\{a\}$, and $\emptyset$, which correspond respectively to the labelings (1), (2), and (3). The set $\{c\}$ is a subset of a completely acceptable set, but is not acceptable. This is due to the fact that in order to show that $\{c\}$ is valid, one must accept $\{a\}$.

Thus, acceptable sets, although they do reflect local validity, require the observer to take into account the context of the argumentation framework.

The following definitions and lemma will allow us to show in theorem 2 that every acceptable set can be completed to form a completely acceptable set.

Definition 5 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let $S \subseteq A$.*

- *The argumentation framework $AF|_S = (S, \rightsquigarrow|_{S \times S})$ is the restriction of the argumentation framework AF to the set S.*
- *A labeling of $AF|_S$ is called a* **restricted labeling of** AF*. A complete labeling of $AF|_S$ is called a* **complete restricted labeling of** AF*.*

Definition 6 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. If L is a set of restricted labelings of AF then the partial mapping*

$$\sqcup L : A \to 2^{\{+,-\}}$$

is defined as follows (we use $L|_a$ to denote the set $\{l \in L \mid l(a)$ is defined $\}$)

$$(\sqcup L)(a) = \begin{cases} \cup_{l \in L|_a} l(a) & \text{if } L|_a \neq \emptyset \\ undefined & \text{if } L|_a = \emptyset \end{cases}$$

We denote $\sqcup\{l_1, l_2\}$ as $l_1 \sqcup l_2$.

Lemma 1 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework.*

- *If L is a set of labelings of AF then $\sqcup L$ is a labeling of AF.*
- *If L is a non-empty set of complete labelings of AF then $\sqcup L$ is a complete labeling of AF.*
- *If l_1 is a labeling and l_2 is a complete labeling of $AF|_{l_1^0}$ then $l_1 \sqcup l_2$ is a complete labeling.*

Theorem 2 *Let S be an acceptable set of an argumentation framework AF. There is a completely acceptable set T such that $S \subseteq T$.*

3.2 Defence in the acceptable semantics

In the previous section we have shown that, although the acceptable semantics reflects a certain local validity of sets of arguments, it also implies that the acceptable set of arguments fi t into the context of the large framework, which means that it can be extended to globally valid set of arguments. This property is shared by the admissible semantics, in which one can always consider only maximal admissible sets. So what is the fundamental difference between the admissible and the acceptable semantics?

Recall that, as defi ned in [Dun95], an argument a is **defended** by a set S of arguments iff any argument $b \in A$ which attacks a, is counterattacked by S. In the following example we show that, according to this defi nition of defence, the given argumentation framework has no defendable sets of arguments.

Example 6 Suppose that a medical patient suffering from certain symptoms takes a blood test, and that the results show the presence of a bacteria of a certain category in his blood. There are two types of bacteria in this category, and the blood test does not pinpoint whether the bacteria present in the blood is of type A or of type B. The possible treatment for this patient would be to take antibiotics. As always in this situation, the doctor tries not to prescribe antibiotics if they are not necessary, since they are harmful to the immune system. If she does prescribe antibiotics, the spectrum of the antibiotics is large enough to cover both type-A bacteria and type-B bacteria.

This situation could be modeled by an argumentation framework, where the arguments are rules of the type used in logic programming, and the attack relation used is **undercutting**. As defi ned in [PS96], a rule a undercuts a rule b iff the consequent of rule a states that the antecedent of rule b does not hold. (By stating that the consequent of rule b is not fulfi lled, rule a shows that rule b is not applicable to the case in question.)

Consider the following set of rules:

a = "If the bacteria in the patient's blood is not of type A then it must be of type B."
b = "If the bacteria in the patient's blood is not of type B then it must be of type A."
c = "If the patient does not have a bacterial infection then giving the antibiotics to the patient is superfluous."
d = "If it is not superfluous to give the patient antibiotics then the antibiotics should be prescribed."

The argumentation framework generated by this set of arguments and the undercutting attacking relation, is shown in figure 7. We have portrayed several of the labelings of this argumentation framework. In each labeling the accepted nodes, which are in l^+, represent arguments which are applied by the reasoner.

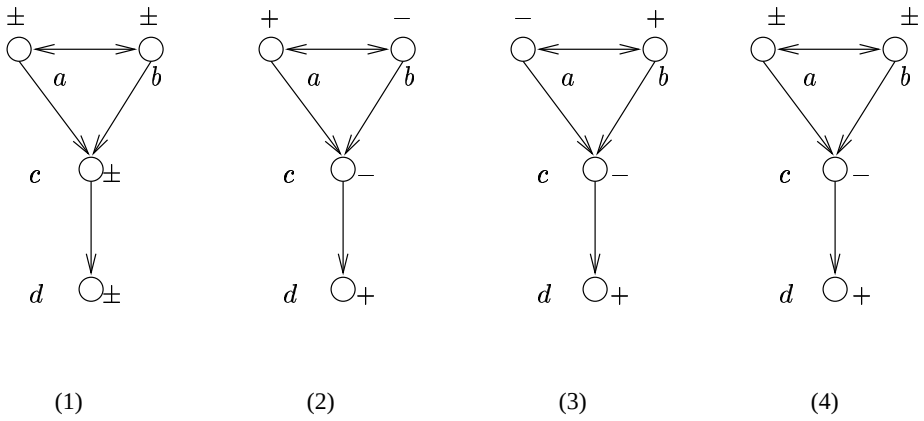


Figure 7: Some labelings for example 6

The labelings (2) and (3) represent the credulous reasoners that guess which bacteria is in the patient's blood and prescribe the antibiotics. The labeling (1) represents the very sceptical reasoner that says "I don't know which bacteria is in the patient's blood, so I'm not taking any decisions". However, it seems unreasonable to allow the uncertainty regarding the type of bacteria to prevent the doctor from taking the decision to prescribe the antibiotics since, whichever one of the two types of bacteria is present, the antibiotics is necessary. It is therefore the labeling (4) which seems the most reasonable, since it represents the reasoner that says "I don't know which bacteria is present, but I know that the argument c is not applicable, since at least one of the bacteria are present; therefore, I should prescribe the antibiotics". Notice that in the labeling (4) the set $\{d\}$ is "defended" by the set $\{a, b\}$, although neither a nor b are accepted in that labeling.

The admissible semantics suggested in [Dun95] adopts only the very sceptical reasoning of labeling (1), and the very credulous reasoning of labelings (2) and (3). It does not adopt the intermediate reasoning of labeling (4); indeed, the set $\{d\}$ is not admissible, since the attack from the argument c is not counterattacked by any argument in this set.

Note that the only way to obtain the set $\{d\}$ from the admissible semantics is to pinpoint all of the credulous sets and to take their intersection. Our acceptable semantics, on the other hand, presents all of the above types of reasoning in one definition, and without requiring that all credulous sets be computed in order to compute a sceptical set. In addition, in section 7 we present a refinement of the acceptable semantics which pinpoints the labeling (4) as the best solution.

The reasoning in the above example shows that, according to the acceptable semantics, an argument can be defended by a set of arguments, even if the defending arguments are not accepted. This type of defence also occurred in the argumentation framework shown in figure 3, which corresponds to the stable-tennis-doubles problem of example 2. According to the labeling (4) in figure 3, the argument e is accepted, not because it is defended by an acceptable argument, but because it is defended by a set of arguments ($\{a, b, c\}$) which, although it contains within itself some uncertainty, eliminates the attacks on e.

With this motivation, we relax [Dun95]'s strict interpretation of defence, where an argument can be used as a defence only if it is itself accepted. In particular, we do not demand consistency of the defence.

Definition 7 *Let $(A, \rightsquigarrow)$ be an argumentation framework. A set $T \subseteq A$ of arguments is a* **supporting defence** *of a consistent set of arguments S iff the following conditions hold:*

1. $\forall a \in A$ *if* $a \rightsquigarrow T \cup S$ *then* $T \cup S \rightsquigarrow a$.
2. $S \not\rightsquigarrow T$.
3. $T \not\rightsquigarrow S$.
4. $S \cap T = \emptyset$.

A consistent set S of arguments in AF is **defendable** *iff there is a set T which is a supporting defence of S.*

Naturally, a basic requirement for a set S to be defendable is, as ensured by condition 1), that any attack on itself or on its defence be countered. Notice that we allow a limited degree of inconsistency in the defence supporting a set S of conclusions since, according to definition 7, we could have $T \rightsquigarrow T$. We do, however, demand consistency within the set S of conclusions and, as conveyed in conditions 2) and 3), we insist that the set S not attack its own defence, nor vice versa.

Example 7 In the argumentation framework of figure 3, the empty set is a supporting defence of the empty set, the set $\{a, b, c\}$ is a supporting defence of the set $\{e\}$, and the set $\{a, b, c, d\}$ is a supporting defence of the set $\{f\}$. Thus, the sets $\emptyset$, $\{e\}$ and $\{f\}$ are each defendable. Notice that the supporting defences $\{a, b, c\}$ and $\{a, b, c, d\}$ are not consistent.

Since the empty set is a supporting defence of any admissible set, we have that

Corollary 1 *Any admissible set is defendable.*

Lemma 2 *Let S be a defendable set of arguments. Let T and T' be supporting defences of S. Then $T \cup T'$ is a supporting defence of S.*

Lemma 2 motivates the following definition:

Definition 8 *Let $(A, \rightsquigarrow)$ be an argumentation framework. The maximal supporting defence of a defendable set $S \subseteq A$ of arguments is the union of the supporting defences of S.*

The correspondence between acceptable sets and defendable sets is given in the following theorems.

Theorem 3 *Let S be an acceptable set of an argumentation framework $AF = \{A, \rightsquigarrow\}$, and let l be a labeling that corresponds to S. The set S is defendable and $l^{\pm}$ is a supporting defence of S.*

Theorem 4 *Let S be a maximal defendable set of an argumentation framework $AF = (A, \rightsquigarrow)$ (i.e. S is defendable and there is no defendable set S' such that $S \subset S'$) and let T be the maximal supporting defence of S. The set S is completely acceptable and corresponds to a complete labeling l such that $l^{\pm} = T$.*

Notice that in order to establish symmetry between the notions of defendability and complete acceptability, there is an additional maximality condition in theorem 4. This is due to the fact that these two semantics belong to the two different philosophies of argumentation which we mentioned earlier; complete acceptability concerns global validity, while defendability, like acceptability, concerns local validity.

The following example shows that the converse of theorem 3 is not true, and that the maximality condition in theorem 4 is necessary:

Example 8 Consider the argumentation framework shown in figure 6, page 10. The set $S = \{c\}$ is defendable, since the set $\{a\}$ is a supporting defence of S. However, S is not acceptable. Notice that S is not maximal defendable since the set $\{a, c\} \supseteq S$ is defendable.

The following example shows that the converse of theorem 4 does not hold:

Example 9 Consider again the argumentation framework shown in figure 5, page 9. The empty set is completely acceptable, as is illustrated by the complete labeling (1). While it is defendable, it is not maximal defendable, since the sets $\{a\}$ and $\{b\}$ are also defendable.

3.3 Self-defeating arguments

The term "self-defeat" is used in different ways in the literature (see [Pol94] and [PS96]). A self-defeating argument in our terminology is simply an argument that attacks itself. This type of contradictory argument has been given a special status in some formalisms (see for example [Pol94]). [Dun95] and [BDKT97] express the desire to extend their theories in order to deal with this special case. In contrast, in our

theory, since the defining rules of definition 3 govern just single (parts of) labels, i.e. "+"s or "−"s, concepts such as contradiction or self-defeat, become *derived concepts*, and there are no special principles or exceptions that need to be defined in order to deal with them. They are handled naturally by the underlying principles (definition 3).

Example 10 Consider the argumentation framework in figure 8. The argument a in

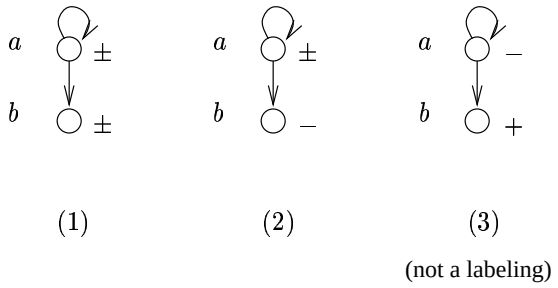


Figure 8: A self-defeating argument

figure 8, since it attacks itself, is a self-defeating argument. A labeling for this argumentation framework, according to [BDKT97], [Dun95] and [Pol94], would assign ± to all nodes[3], as in the labeling (1). Definition 3 provides this as well but it also supports the labeling (2). Note that the mapping (3) is not legal, since the argument a has no supported attacker, and thus condition 1 of definition 3 has been violated.

Intuitively, labeling (2) can be motivated by noting that, since a is "strong enough" to attack (i.e. add a "−" to the label of) a, surely it is also strong enough to do the same with b.

The following example supports our intuition about self-defeating arguments using arguments derived from logic programs.

Example 11 Consider the following logic program:

$$\begin{array}{lcl} p & \leftarrow & \neg p \\ q & \leftarrow & \neg p \end{array}$$

A simple mapping (see [JV96]) translates proof trees into arguments. In the associated

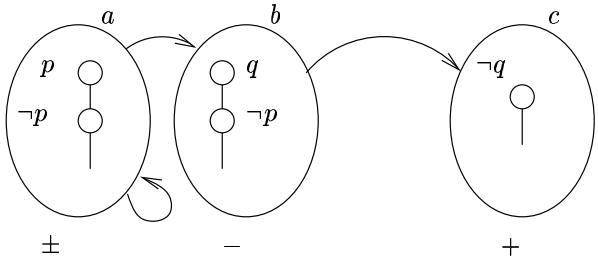


Figure 9: Mapping logic programs to frameworks

[3] after suitably mapping their approach to the present framework (see also section 6.3)

argumentation framework (figure 9), one proof tree attacks another if its conclusion contradicts some leaf node (labeled by a negative literal) of the other. The labeling of figure 9 then corresponds to the 3-valued interpretation $\{\neg q\}$, since it assigns "$-$" to the only proof tree for q.[4] The interpretation $\{\neg q\}$ is intuitive since there is no way that q can be founded (using $\neg p$), as that would lead to a contradiction. Therefore, using negation as failure, $\neg q$ may be accepted.

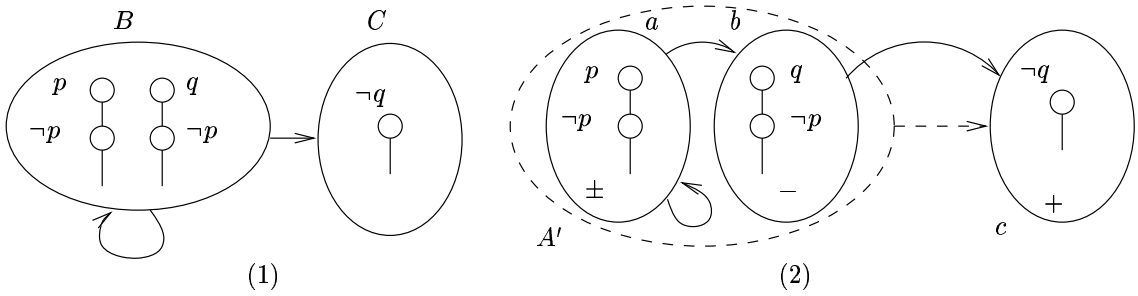


Figure 10: Another way to map logic programs to frameworks

The above mapping has also been discussed in [Dun95]. [KMD94] (see also section 6.2) also maps logic programs onto argumentation frameworks, with a different mapping than the one used here. [KMD94] also proposes the interpretation $\{\neg q\}$ for a similar logic program in which the self-defeat takes the form of an odd loop. Essentially, [KMD94] considers arguments to be *sets* of proof trees, as illustrated in the first part of figure 10. Note that [KMD94] accepts the argument C while rejecting B. The second part of figure 10 hints[5] at how our mapping (and labeling) can be related to [KMD94]'s: [KMD94]'s single self-defeating argument B can be regarded as a "summary" of a subframework A' consisting of $\{a, b\}$ in figure figure 10 (2). Note that the summary is labeled "$-$" because the interface between A' and the rest of the framework consists entirely of rejected arguments (b in the example). On the other hand, in the summary view the label "$-$" on A' is "internally motivated" and thus condition 1 in definition 3 need not be enforced.

Our intuition about self-defeat fits naturally with other, less controversial frameworks:

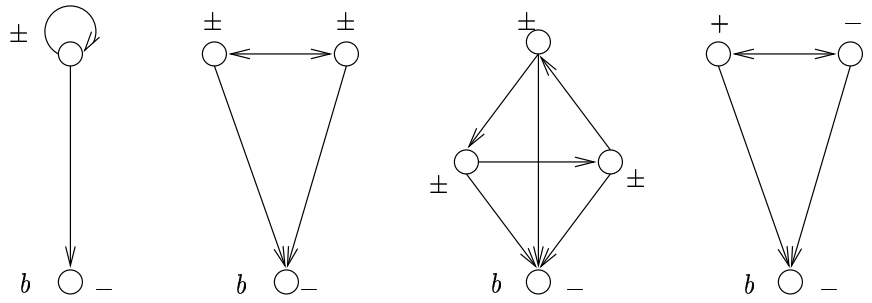
Example 12 The labelings of the argumentation frameworks shown in figure 11 are consistent, in the sense that the framework (1) can be regarded as a special case of both (2) (which was discussed in example 6) and (3). Figure 11 also illustrates how arguments may themselves consist of sub-frameworks[4] and, in that sense, (1) can be regarded also as a "summary" of (4).

The following example demonstrates some consequences of our approach:

Example 13 Consider again the argumentation framework of figure 2. The arguments a, b and c are in a situation which is similar to self-defeat, since they are caught in an odd loop. This example illustrates the propagation of contradiction in argumentation frameworks. When there is ambiguity about one or several arguments, as is the

[4] Obviously, a "$+$" node accepts the conclusion of the proof tree, a "$-$" node accepts its negation, while $\pm$ makes it undefined.

[5] This is a topic for further research, see section 7.



(1) (2) (3) (4)

Figure 11: Equivalent argumentation frameworks

case for the arguments a, b and c in the example, uncertainty can spread throughout the framework, eliminating the possibility to decide on the validity of other arguments. This occurs in the labeling (1), which reflects the occurrence of this phenomenon in the [BDKT97] and [Dun95] semantics. This problem becomes acute in large argumentation frameworks, since contradiction is likely to appear somewhere in a large framework. The labeling (4), on the other hand, allows the minimization of the effects of contradiction.

4 Refinements of the acceptable semantics

In this section we present various refinements of the acceptable semantics, which are each appropriate for different applications.

4.1 The minimal semantics

In this section we introduce a global semantics for argumentation frameworks which is sceptical, in that it limits the set of accepted arguments to those arguments which are beyond dispute. It is defined by means of the so-called "minimal" complete labeling. This complete labeling shall be defined with the help of transfinite sequences (see [Men79]), i.e. sequences on the class On of ordinal numbers, as follows:

Definition 9 *Let $(A, \rightsquigarrow)$ be an argumentation framework. Define the transfinite sequences $S_\alpha, T_\alpha, \alpha \in On$ as follows: let*

$$S_0 = T_0 = \emptyset$$

For any successor ordinal α, define

$$\begin{aligned} S_\alpha &= \{a \in A \mid \forall b \rightsquigarrow a : b \in T_{\alpha-1}\} \\ T_\alpha &= \{a \in A \mid S_\alpha \rightsquigarrow a\} \end{aligned}$$

For any limit ordinal $\alpha > 0$, define

$$S_\alpha = \bigcup_{\beta<\alpha} S_\beta$$
$$T_\alpha = \bigcup_{\beta<\alpha} T_\beta$$

Clearly, the sequences S_α and T_α are monotonic with respect to set inclusion so each sequence has a respective least fixpoint. Define S as the least fixpoint of the sequence S_α,$\alpha \in On$, and T as the least fixpoint of the sequence T_α, $\alpha \in On$.

The following lemma allows us to define the mapping of definition 10:

Lemma 3 *Let $(A, \rightsquigarrow)$ be an argumentation framework. The fixpoints S and T, defined in definition 9, are disjoint.*

Definition 10 *Let $(A, \rightsquigarrow)$ be an argumentation framework. Define the mapping l_{min} as follows: $\forall a \in A$, let*

$$l_{min}(a) = \begin{cases} + & \text{if } a \in S \\ - & \text{if } a \in T \\ \pm & \text{otherwise} \end{cases}$$

Theorem 5 *The mapping l_{min} defined in definition 10 is a complete labeling.*

We shall show in theorem 6 that if an argument a is accepted in the above complete labeling (i.e. $l_{min}(a) = +$), then it is accepted in any complete labeling. Similarly, if an argument a is rejected in the above complete labeling (i.e. $l_{min}(a) = -$), then it is rejected in any complete labeling. This motivates the following terminology:

Definition 11 *Let $AF = A, \rightsquigarrow$ be an argumentation framework.*

- *The mapping l_{min} defined in definition 10 is called the* **minimal complete labeling** *of AF.*
- *A set $S \subseteq A$ of arguments is said to be the* **minimal set** *iff it corresponds to the minimal complete labeling.*

Example 14 Consider again the decision-making problem presented in example 4, which concerns deciding what to believe when two equally reliable friends provide two contradictory pieces of information. The minimal labeling is the one shown in labeling (1) of figure 5. Thus, the minimal semantics corresponds to the sceptical reasoner that hears both arguments and withholds belief.

Example 15 Consider again the medical problem presented in example 6. The minimal semantics corresponds to the sceptical reasoner who says "I don't know which bacteria is in the patient's blood, so I'm not taking any decisions". Thus, the minimal labeling is the one shown in labeling (1) of figure 7.

It should be noted that the minimal set is essentially the well-founded set of [BDKT97], the grounded set of [Dun95], and the set of justified arguments of [PS96]. However, the description of this model in terms of labelings allows for elegant manipulation and comparison of the various models, as in the following definition:

Definition 12 *Let l_1 and l_2 be labelings of an argumentation framework $AF = (A, \rightsquigarrow)$. l_1 is called a* **refinement** *of l_2, denoted $l_1 \sqsubseteq l_2$, iff $\forall a \in A, l_1(a) \subseteq l_2(a)$. If $l_1 \sqsubseteq l_2$ we also say that (the set corresponding to) l_1 is more* ***credulous*** *than (the set corresponding to) l_2; or, alternatively, that (the set corresponding to) l_2 is more* ***sceptical*** *than (the set corresponding to) l_1.*

Intuitively, a more credulous refinement $l_1 \sqsubseteq l_2$ has less undecided ($\pm$) arguments than l_2.

The following theorem shows that the minimal semantics is sceptical.

Theorem 6 *Any labeling of an argumentation framework $AF = (A, \rightsquigarrow)$ is a refinement of the minimal complete labeling of AF. In particular, any argument $a \in A$ which is accepted in the minimal complete labeling (i.e. $l_{min} = +$) is accepted in any complete labeling. Similarly, any argument $b \in A$ which is rejected in the minimal complete labeling (i.e. $l_{min} = -$) is rejected in any complete labeling.*

The following lemma refers to the operation $\sqcup$ for combining labelings, which was defined in definition 6:

Lemma 4 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. Let L be a set of refinements of a labeling l of AF. The labeling $\sqcup L$ is a refinement of l.*

Corollary 2 *Let $\mathcal{L}$ be the set of all labelings of an argumentation framework AF. $\mathcal{L}_{\sqsubseteq}$ is a complete lattice, i.e. $\sqsubseteq$ is a partial order and for any $S \subseteq \mathcal{L}$, there are labelings $lub(S)$ and $glb(S)$. Furthermore, $lub(S) = \sqcup S$, and the top element of $\mathcal{L}_{\sqsubseteq}$ is the minimal complete labeling of AF.*

Notice in the corollary that the lowest upper bound of a set of labelings can be found by taking the union of its symbols, denoted by the operation $\sqcup$. At first glance, one might think that the greatest lower bound could be found by taking the intersection of its symbols. However, the intersection of two labelings is not necessarily a labeling, as shown by the following example:

Example 16 Consider the argumentation framework $AF = (A, \rightsquigarrow)$ of figure 12. Let the operation $\sqcap$ be defined as follows: $\forall a \in A, l_1 \sqcap l_2(a) = l_1(a) \cap l_2(a)$. In the figure, l_1 and l_2 are both labelings of the argumentation framework. Their "intersection" $l_1 \sqcap l_2$, however, is not a labeling, since it violates condition 1 of definition 3. Their greatest lower bound is the labeling which assigns the empty set to each argument in the framework.

Notice that corollary 2, which concerns (partial) labelings, has no equivalent for the set of complete labelings. In fact, while the set of labelings, ordered by $\sqsubseteq$, forms a complete lattice, the set of complete labelings does not even form a lattice. Indeed, the complete labelings l_1 and l_2 of figure 12 have no lower bound in the set of complete labelings. In addition, if S is empty then $\sqcup S$ is not a complete labeling.

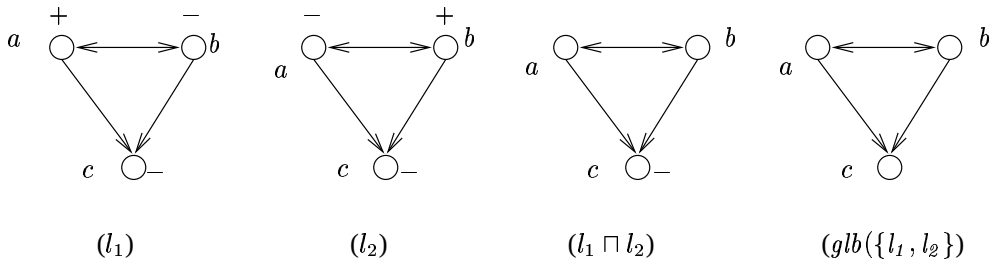


Figure 12: The intersection of two labelings is not necessarily a labeling

4.2 The robust semantics

4.2.1 Specification for the robust semantics

In section 3 we demonstrated that an argumentation framework might have several distinct labelings. In section 4.1 we pointed out a particular complete labeling which corresponds to a sceptical global semantics. In section 4.2.3 we shall define a global and local semantics which is both credulous and sceptical, by restricting ourselves to a certain category of labelings which we shall call "robust". In this section we provide a specification for the robust semantics, which gives the motivation behind the definition of section 4.2.3.

The idea of the robust semantics is to choose those labelings that respect the stability of the decided arguments. When a set of conclusions is proposed, the so-called "undecided part" of the argumentation framework, consists of the arguments which have not yet been decided (i.e. the arguments in $l^{\pm}$), with their interactions. The information contained in this restricted argumentation framework may permit further conclusions to be added to the decided arguments (i.e. the arguments in $l^{+} \cup l^{-}$). These additional decisions are expressed by a complete labeling of the undecided part of the argumentation framework. In this case, the original set of arguments may or may not be compatible with the newly added conclusions, in that their combination may or may not correspond to a labeling. We shall say that a labeling l is *robust*, if the decided arguments can remain unchanged, even if some undecided arguments become decided.

Example 17 Consider again the medical decision-making problem presented in example 6. We discussed the types of reasoners that corresponded to various labelings of the argumentation framework associated to the problem. Notice that there is an additional labeling of the argumentation framework of figure 7, which we did not portray in the figure. This labeling l is portrayed in figure 13. The labeling l corresponds to the reasoner that says "I don't know which of the bacteria are present in the patient's blood, so I won't decide whether or not the antibiotics are superfluous. Since antibiotics might be superfluous I will not prescribe them. Thus, this reasoner says "I don't know which of the arguments a and b are applicable, so I can't decide if the argument c is applicable. Since I can't accept c, I will reject d".

The reasoner in question delays deciding about c, because of the uncertainty regarding a and b. But he does decide to reject d. His reasoning is not sceptical since, although there is uncertainty as to which of a and b is true, he does exclude d. Neither is his reasoning credulous, since he withholds from deciding about c. His reasoning is an unhappy medium between the two. Why is it faulty? Suppose that, after taking the

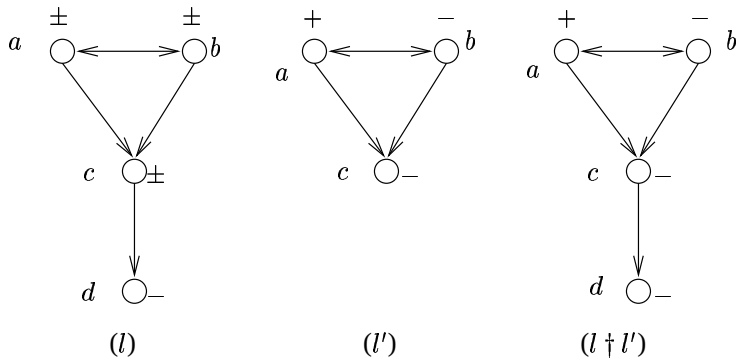


Figure 13: l is not robust

decision to reject d, he reconsiders the situation by reviewing the arguments which he left undecided. Suppose that it then becomes apparent that it is the bacteria A that is present in the patient's blood. The reasoner then has to admit that his exclusion of d was unjustified. Formally, this means that the labeling l which corresponds to his original reasoning, is not compatible with the restricted labeling l', which corresponds to the added information that concerns the undecided arguments. In the figure, the incompatibility of these two labelings is shown by the fact that their combination $l \dagger l'$, which contains each of their decided arguments, is not a labeling. Indeed, the first condition of definition 3 is not satisfied at the node d.

The notions to which we have referred in this example, shall be formalized in the coming definitions. The following notion conveys the idea of extending a present knowledge state, by adding further conclusions which concern undecided arguments:

Definition 13 *Let l be a labeling of an argumentation framework $AF = (A, \rightsquigarrow)$. A complete labeling of $AF|_{l^\pm} = (l^\pm, \rightsquigarrow|_{l^\pm \times l^\pm})$ is said to be an* **extension** *of l.*

The following notion of *compatibility* is a criterion which determines whether the combination of two successive sets of decisions forms one coherent set of decisions:

Definition 14 *Let l be a labeling of an argumentation framework $AF = (A, \rightsquigarrow)$, and let l' be an extension of l. The labeling l is said to be* **compatible** *with l' iff $l \dagger l'$ is a labeling of AF, where $l \dagger l'$ is defined as follows:*

$$l \dagger l'(a) = \begin{cases} l'(a) & \text{if } a \in l^\pm \\ l(a) & \text{otherwise} \end{cases}$$

In order for a set of decisions to be considered "good", it must be compatible with any "good" extension of itself. Thus, any extension which is not compatible must be itself invalidated by a further extension which is valid. This motivates the following specification for robust labelings:

Specification 1 *Let AF be an argumentation framework.*

- *A labeling l of AF is* **robust** *iff*

1. *it is compatible with all robust extensions of itself, and*
2. *any incompatible extension l' of l has a robust extension which is incompatible with l'.*[6]

- *A set $T \subseteq A$ of arguments is said to be* **robust** *iff it corresponds to a robust labeling of AF.*
- *A set $T \subseteq A$ of arguments is said to be* **completely robust** *iff it corresponds to a robust complete labeling of AF.*

Example 18 Consider again the stable-tennis-doubles problem presented in section 2. Recall that in that application a labeling represents an arrangement of pairs of players into stable pairs (which are accepted), rejected pairs, and undecided pairs. Suppose then that more pairs are needed, and that there is therefore an attempt to categorize some of the undecided pairs into accepted or rejected pairs. This further arrangement of pairs is an *extension* to the original arrangement. If the original arrangement is robust then the two successive sets of stable pairs can simply be joined together, and their union corresponds to a satisfactory arrangement of pairs. If, on the other hand, the original arrangement is not robust, then the union of the two sets of stable pairs might not correspond to a satisfactory arrangement. In this case, when the original arrangement is extended, some of its stable pairs are no longer stable, and have to change their status in the categorization. This "instability" of the pairs in the original arrangement seems to undermine the motivation behind the definition of stable pairs. It seems therefore reasonable to require arrangements to be robust.

In particular, consider the argumentation framework which corresponds to the stable-tennis-doubles problem, shown in figure 3. The complete labeling (4) is robust, since the only complete labeling of the argumentation framework restricted to the set $\{a, b, c\}$ is the one given. Thus, there is no non-trivial extension of this labeling. Similarly, the complete labeling l' of the restricted argumentation framework $AF|_{\{a,b,c,d\}}$ shown in figure 14 is robust. The complete labeling (3) in figure 3, which we represent as the complete labeling l in figure 14, is not robust since it is not compatible with its extension l'. Indeed, the mapping $l \dagger l'$ violates condition (1) of definition 3 at the node e, so it is not a labeling.

In terms of the corresponding stable-tennis-doubles problem, this means that the stable pair $\{Rafter, Chesnokov\}$ which corresponds to l, although it corresponds to a solution of the stable-tennis-doubles problem, is not a satisfactory solution since, when it is extended, it is no longer stable.

4.2.2 The meta-argumentation framework

Recall in example 18 that the arrangement represented by labeling l in figure 14 was rejected by the robust semantics. This occurred because the undecided pairs could be arranged in a way which was incompatible with the matched pairs. This further arrangement of the undecided pairs was represented by a restricted labeling l'. Since it is l' that invalidates l, the restricted labeling l' can be seen as a threat to the labeling l. Similarly, all incompatible extensions of l can be seen as attacks on l. This suggests

[6] The reader who is suspicious of the apparent circularity in specification 1 is referred to section 4.2.3, where a proper definition of robustness will be presented.

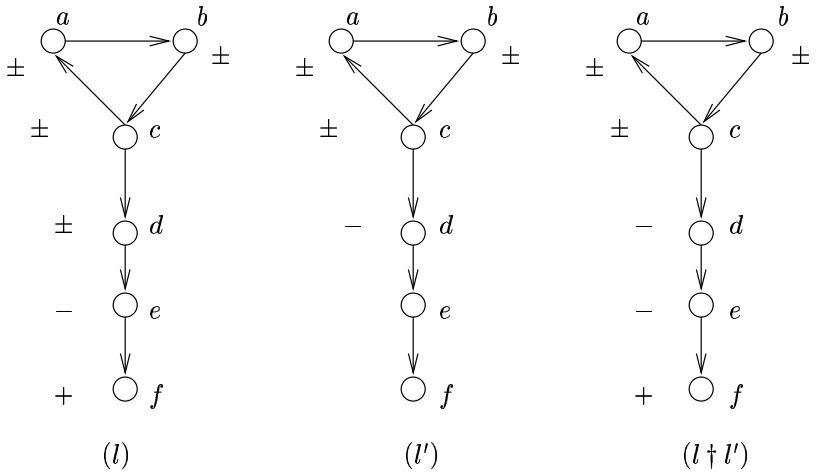


Figure 14: l is not robust

considering the set of restricted labelings as a set of arguments, and the incompatibility of extensions as an attacking relation. This naturally defines an argumentation framework as follows:

Definition 15 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The* **meta-argumentation framework** *of AF is the argumentation framework $AF^* = (A^*, \rightsquigarrow^*)$, where*

- *A^* is the set of restricted labelings of AF, i.e.*

$$A^* \equiv \{l \mid l \text{ is a labeling of } AF|_S \text{ for some } S \subseteq A\}$$

- *$\forall l, l' \in A^*$, $l' \rightsquigarrow^* l$ iff l' is an incompatible extension of l.*

Example 19 The argumentation framework shown in figure 15 has five complete labelings, and four additional labelings. In addition, there are many restricted labelings. Among these numerous restricted labelings, there are four that, in the meta-argumentation framework, attack some of the non-restricted labelings of figure 15. In figure 16 we depict those four restricted labelings. All of the labelings and restricted labelings of AF, together with their attacks, are represented in the meta-argumentation framework. In figure 17 we represent the important part of the meta-argumentation framework, which is generated by the labelings of AF and the four restricted labelings of figure 15, together with their attacks. (The remaining restricted labelings, since they do not attack nor are attacked in the meta-argumentation framework, are isolated nodes in the meta-argumentation framework which do not represent labelings and do not affect any of the labelings. For clarity we have omitted them from figure 17.)

We now show how to express the robust specification in terms of the meta-argumentation framework. It is obvious that the following specification for robust labelings is equivalent to the one in specification 1:

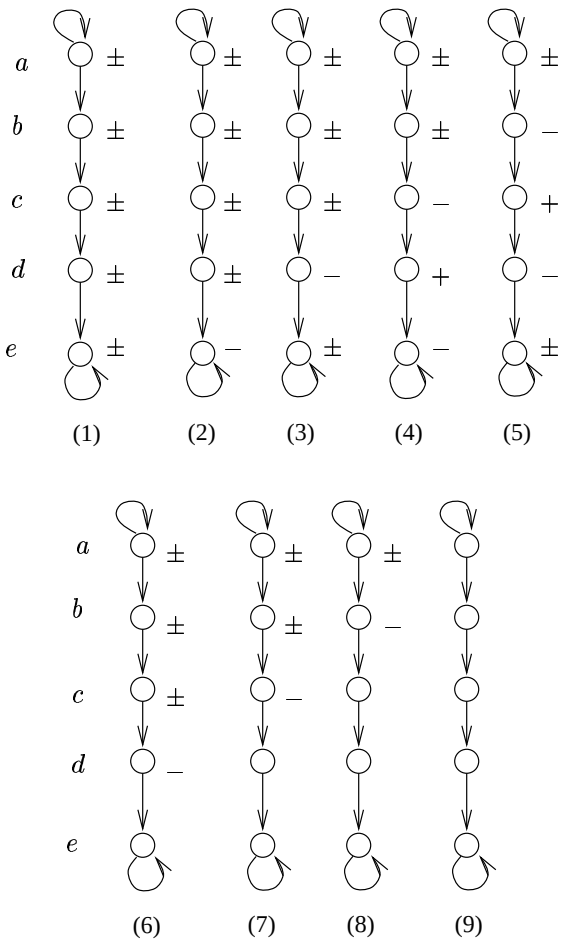


Figure 15: Labelings underlying the meta-framework of figure 17

Specification 2 *Let l be a labeling of an argumentation framework $AF = (A, \rightsquigarrow)$. The labeling l is robust iff $\forall l' \rightsquigarrow^* l$, l' is not robust and $\exists l'' \rightsquigarrow^* l'$ such that l'' is robust.*

In the following section we will refer to the attacking relation $\rightsquigarrow^*$ to define the robust semantics. In section 4.2.4 we will show that robustness of a labeling is a simple property of the associated argument in the meta-argumentation framework.

4.2.3 Fixpoint definition of the robust semantics

In section 4.2.1 we gave a recursive specification of the robust semantics. In this section we provide a fixpoint definition of the robust semantics, and show that the robust semantics satisfies its specification. The definition presented in this section uses the meta-argumentation framework defined in definition 15.

Consider the following function, which is naturally associated to specification 2:

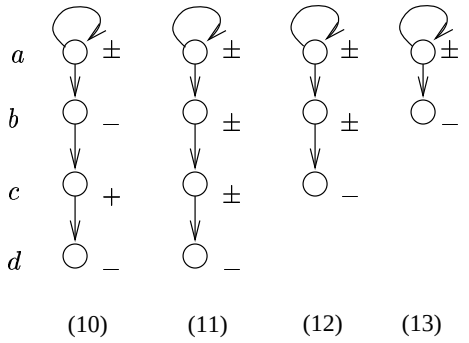


Figure 16: Some restricted labelings underlying the meta-framework of fi gure 17

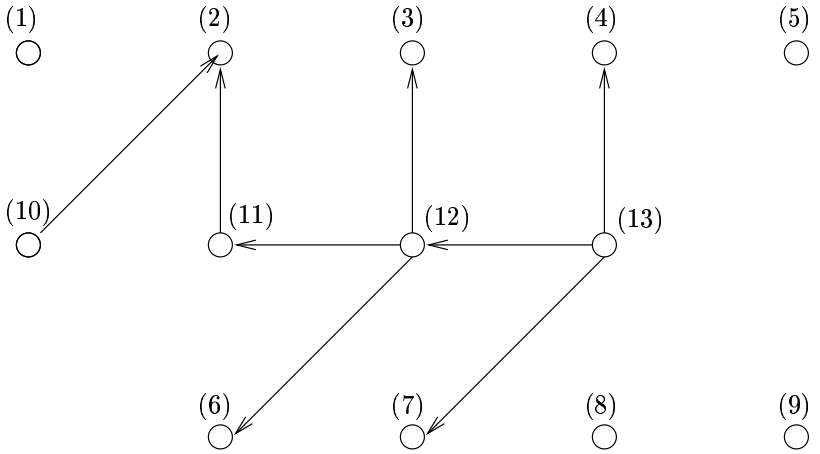


Figure 17: The (main part of the) meta-framework of the framework of fi gure 15

Definition 16 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let $AF^* = (A^*, \rightsquigarrow^*)$ be the meta-argumentation framework of AF. Define the following function N:*

$$\begin{aligned} N : 2^{A^*} &\rightarrow 2^{A^*} \\ N(X) &\equiv \{l \in A^* \mid \forall l' \rightsquigarrow^* l, l' \notin X \wedge \exists l'' \rightsquigarrow l' \textit{ such that } l'' \in X\} \end{aligned}$$

Although N is not necessarily monotonic, it does have a well-defi ned and unique least fi xpoint, as shown by the following lemma:

Lemma 5 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The function N, defined in definition 16, has a least fixpoint.*

This lemma allows us to defi ne the robust semantics as follows:

Definition 17 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. A restricted labeling l of AF is robust iff it is in the least fixpoint of the function N.*

The robust semantics, defi ned in defi nition 17, indeed corresponds to its specifi cation, as shown by the following theorem.

Theorem 7 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The set of robust restricted labelings satisfies specification 2, i.e. if X is the set of robust restricted labelings then $X = \{l \in A^* \mid \forall l' \rightsquigarrow^* l, l' \notin X \wedge \exists l'' \rightsquigarrow l' \text{ such that } l'' \in X\}$.*

4.2.4 The robust semantics as the minimal semantics of the meta-argumentation framework

Since the meta-argumentation framework is at the basis of the robust semantics, there is reason to suspect that the robust semantics of an argumentation framework can be read from its meta-argumentation framework. In addition, since the meta-argumentation framework is itself an argumentation framework, it seems natural to explore its semantics. The simplest semantics which we have presented so far is the minimal semantics. It is the most straightforward global semantics and, as we have already demonstrated, it is the most natural sceptical semantics. In the following example we show that the minimal semantics of the meta-argumentation framework associated to the argumentation framework discussed in example 19 produces the robust labelings of this argumentation framework.

Example 20 Consider the argumentation framework shown in figure 15. The main part of its meta-argumentation framework is shown in figure 17. The minimal labeling of the main part of the meta-argumentation framework is shown in figure 18. Observe

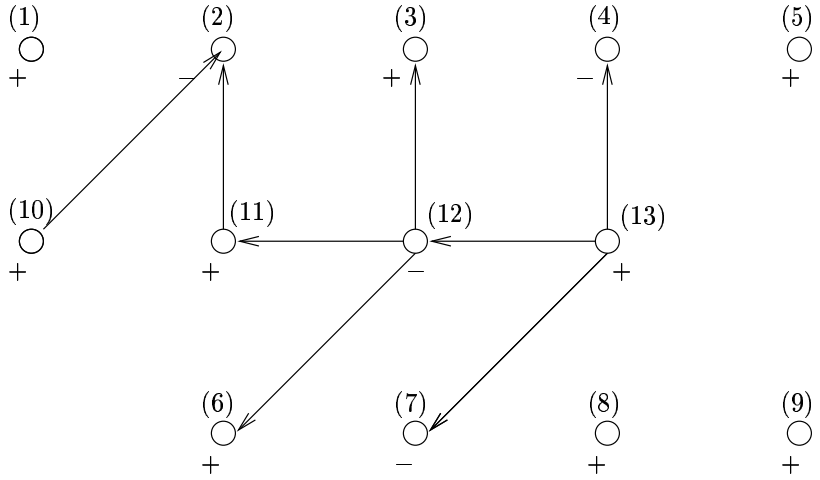


Figure 18: The minimal labeling of the meta-framework of example 20

that this labeling corresponds exactly to the robust (restricted) labelings of the framework of figure 15.

This example motivates the following definition:

Definition 18 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework and let $AF^* = (A^*, \rightsquigarrow^*)$ be its meta-argumentation framework. Define S^* and T^* as the fixpoints of the sequences defined in definition 9, with respect to the argumentation framework AF^*. Thus, let*

$$S_0^* = T_0^* = \emptyset$$

For any successor ordinal α, define

$$\begin{aligned} S^*_\alpha &= \{l \in A^* \mid \forall l' \rightsquigarrow^* l : l' \in T^*_{\alpha-1}\} \\ T^*_\alpha &= \{l \in A^* \mid S^*_\alpha \rightsquigarrow^* l\} \end{aligned}$$

For any limit ordinal $\alpha > 0$, define

$$\begin{aligned} S^*_\alpha &= \bigcup_{\beta<\alpha} S^*_\beta \\ T^*_\alpha &= \bigcup_{\beta<\alpha} T^*_\beta \end{aligned}$$

and

$$S^* = \textit{the least fixpoint of the monotonic sequence } S^*_\alpha, \alpha \in On \quad (1)$$

$$T^* = \textit{the least fixpoint of the monotonic sequence } T^*_\alpha, \alpha \in On. \quad (2)$$

The following lemma provides a constructive way of determining the robust semantics:

Lemma 6 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework and let $AF^* = (A^*, \rightsquigarrow^*)$ be its meta-argumentation framework. The set S^* is the least fixpoint of the function N, and is therefore the set of robust restricted labelings.*

Theorem 8 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let l^*_{min} be the minimal complete labeling of $AF^* = (A^*, \rightsquigarrow^*)$. The restricted labelings that are labeled + are the robust restricted labelings of AF, i.e.*

$${l^*_{min}}^+ = \{l \in A^* \mid l \textit{ is robust }\}.$$

Corollary 3 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let l^*_{min} be the minimal complete labeling of AF^*. The set of robust labelings of AF is the set $\{l \in {l^*_{min}}^+ \mid$ the domain of l is $A\}$.*

Thus, we have shown that the robust semantics is given simply by determining the minimal semantics of an associated argumentation framework. This greatly facilitates the understanding of the robust semantics and the determination of robust sets, since the minimal semantics is the simplest, most straightforward global semantics for argumentation frameworks and, in addition, is sceptical, in that it produces only those sets which *must* be accepted.

5 Relationships between the various semantics

In this section we examine the relationships between the various semantics defined so far. In the following section we will examine the relationships between these and other semantics from the literature.[7]

[7] The results in these sections are valid for any given argumentation framework; we will not always specify the name of the argumentation framework.

For visual clarity, we have divided the results of these two chapters into two separate, but complementary, diagrams. Figure 19 shows how all of the various semantics compare with each other, and supplies references to the proofs of the information contained in the diagram. For those semantics which are shown in figure 19 as incomparable, the references to the proofs appear in figure 20. The reader is encouraged to refer to these two diagrams throughout the reading of sections 5 and 6.

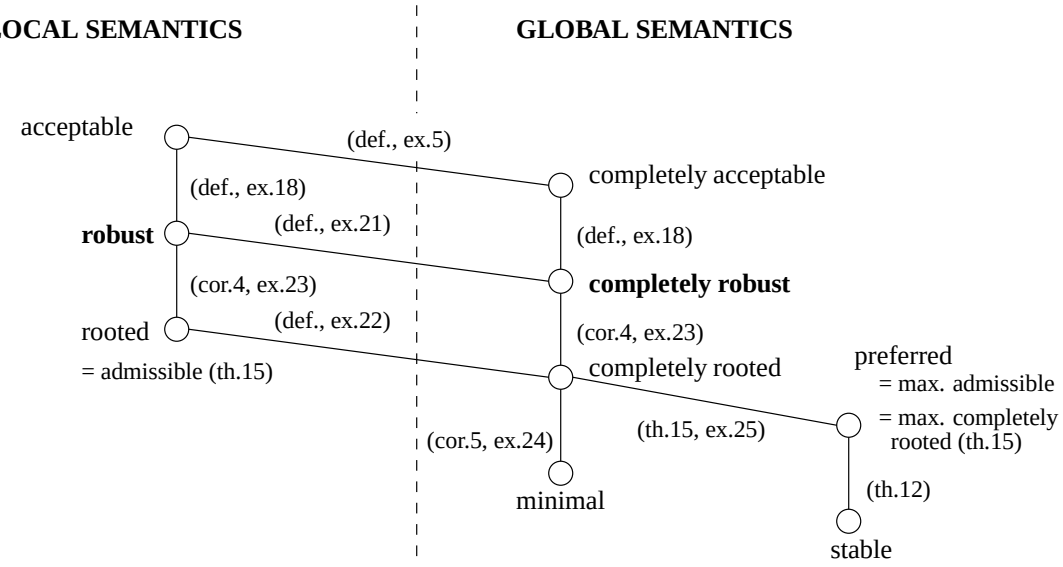


Figure 19: Comparison of various semantics (part 1): a semantics x is strictly included in a semantics y, iff x is lower than y and x is connected to y by a rising edge or a path of rising edges.

As we discussed in section 3.1, there is a distinction between the acceptable and completely acceptable semantics, with respect to their approach to global vs. local validity of sets of conclusions. A local semantics attempts to allow the observer to accept only those arguments that interest her, and to ignore those arguments that do not. In a global semantics something must be said about each argument. All of the semantics which we discuss here are refinements of the acceptable semantics, so we will categorize each of them as global or local.

The semantics which this paper advocates – the robust and completely robust semantics – are, respectively, acceptable and completely acceptable (by definition). The following example shows that robust sets are not necessarily completely acceptable.

Example 21 Consider the argumentation framework consisting of one argument a, with no attacks. The acceptable sets, which are also robust, are the empty set and the set $\{a\}$. The only completely acceptable set is the set $\{a\}$, which is also completely robust.

We now introduce a semantics which will clarify the connections between the other semantics. It is based on so-called *rooted* labelings. This semantics, which is both

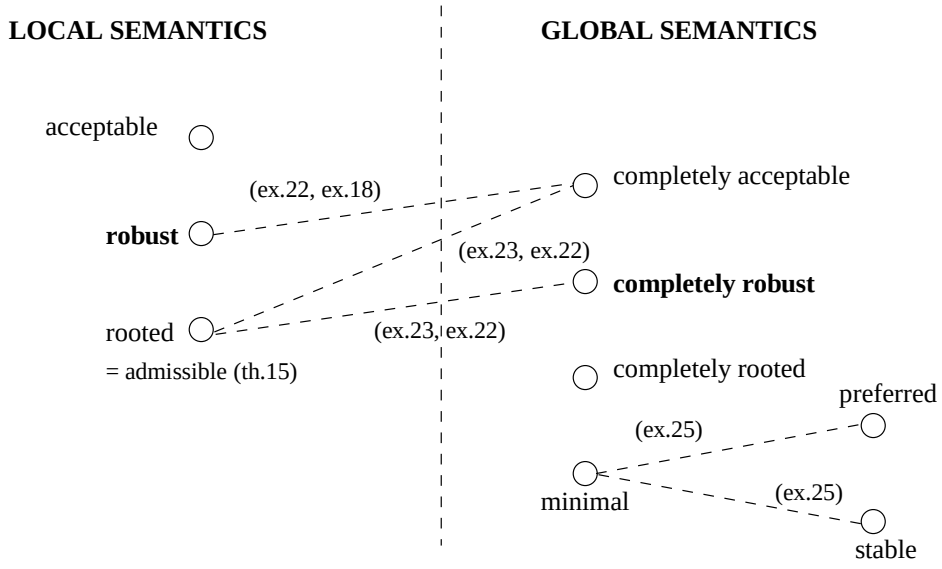


Figure 20: Comparison of various semantics (part 2): a semantics x is not included in, nor includes, a semantics y, iff x is linked to y by a dotted edge.

credulous and sceptical, is realistic in that it insists that in order for a certain set of arguments to be rejected, all of the arguments in the set must be "rooted" to accepted arguments, which means that in this semantics we only reject arguments that are attacked by an accepted argument. We do not reject arguments that are attacked only from undecided arguments. Attacks from undecided arguments are not rooted, since an undecided argument could later become rejected, in which case its attack is no longer potent.

Definition 19 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework.*

- *A labeling l of AF is* **rooted** *iff $\forall a \in l^- \; \exists b \in l^+$ such that $b \rightsquigarrow a$*
- *A set $S \subseteq A$ of arguments is* **rooted** *iff it corresponds to a rooted labeling.*
- *A set $S \subseteq A$ of arguments is* **completely rooted** *iff it corresponds to a rooted complete labeling.*

Notice that the rooted semantics is a local semantics, while the completely rooted semantics is a global semantics.

The following example shows that not all rooted sets are completely rooted:

Example 22 Consider again the argumentation framework consisting of one argument a, with no attacks. The empty set is rooted and robust but, as we pointed out in example 21, is not completely acceptable, and therefore surely does not correspond to any rooted (or robust) complete labeling.

The following lemma and theorem show how to complete a rooted set, to form a completely rooted set. It uses the operation $\sqcup$, which was defined in definition 6, and the fact that, as shall be shown in theorem 11, the minimal complete labeling is rooted.

Lemma 7 *Let l be a rooted labeling of an argumentation framework AF and let l' be the minimal complete labeling of $AF|_{l^\emptyset}$. The complete labeling $l \sqcup l'$ is rooted.*

Theorem 9 *Let S be a rooted set of an argumentation framework AF. There is a completely rooted set T such that $S \subseteq T$.*

Because of the simple definition of the rooted semantics, it is often straightforward to show that a given set of arguments is rooted The following theorem and corollary show that the rooted semantics can then be useful in order to show that the given set is robust.

Theorem 10 *All rooted labelings are robust.*

Corollary 4 *Every rooted set is robust and every completely rooted set is completely robust.*

The following example shows that the converses of theorem 10 and corollary 4 are not true:

Example 23 Consider again the argumentation framework resulting from the medical problem, shown in figure 7, page 12. $\{d\}$ is (completely) acceptable but not rooted not completely rooted.

In addition, examples 22 and 23 showed that the completely robust semantics and the rooted semantics are not comparable. Indeed, example 22 showed that there are rooted sets that are not completely robust, while example 23 showed that there are completely robust sets that are not rooted.

The following theorem, which concerns the sceptical semantics introduced in section 4.1, implies that every argumentation framework has a complete labeling which is robust:

Theorem 11 *The minimal complete labeling is rooted.*

Corollary 5 *The minimal set is completely rooted.*

Corollary 6 *The minimal set is completely robust.*

The following example shows that the converses of theorem 11 and corollary 5 are not true.

Example 24 Consider again the argumentation framework shown in figure 5, page 9. The empty set is the minimal set, while the sets $\emptyset$, $\{a\}$, and $\{b\}$ are the completely rooted sets.

6 Relationships with other approaches

6.1 Comparison of robust semantics with [BDKT97] and [Dun95]

In [BDKT97] the authors propose five different semantics for argumentation frameworks, some of which are based on [Dun95]. They show that several systems for non-monotonic logic – Theorist, default logic, logic programming, auto-epistemic logic, non-monotonic modal logics and circumscription – are instances of these abstract argumentation-theoretical semantics. The five semantics which they suggest – the naive semantics, the stable semantics, the admissible semantics, the preferential semantics and the complete semantics – are all included in the robust semantics which we presented here. On the other hand, the semantics which we advocate in this paper – which is the robust semantics – is not captured in [BDKT97], as we showed in example 10. As we have explained in section 3.3, the present paper and [BDKT97] have different intuitions on how to deal with self-defeating arguments.

Some of the refinements of the robust semantics are considered also in [BDKT97]. For example, our minimal labeling corresponds essentially to the well-founded semantics of [BDKT97] and the grounded semantics of [Dun95]. Furthermore, stable, preferred and admissible sets were studied in [Dun95] and [BDKT97]:

Theorem 12 ([Dun95] and [BDKT97]) *Every stable set is preferred but not every preferred set is stable.*

Theorem 13 ([Dun95] and [BDKT97]) *Every admissible set is preferred but not every preferred set is admissible.*

The following theorem shows how to express stable sets in terms of complete labelings:

Theorem 14 *A set S of arguments in an argumentation framework AF is a stable set iff there is a complete labeling l which corresponds to S such that $l^{\pm} = \emptyset$.*

Intuitively, the idea of this theorem is that if a set of arguments attacks everything else in the framework and is itself consistent, then it is independent enough not to need a supporting defence.

The relationship between the admissible semantics and our semantics is as follows:

Theorem 15 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let $S \subseteq A$ be a set of arguments.*

1. *S is rooted iff S is admissible.*
2. *S is maximal completely rooted (i.e. S is completely rooted and there is no completely rooted set T such that $S \subset T$) iff S is preferred.*

Examples 22 and 25 show that theorem 15 cannot be made stronger; i.e. a set can be admissible and not completely rooted, and a set can be completely rooted and not preferred. This means that the completely rooted semantics is not captured by the admissible semantics or the preferred semantics; rather, it lies in between them.

The following example also shows that the minimal set is not captured by the preferred semantics, nor vice versa.

Example 25 Consider again the argumentation framework shown in figure 5, page 9. As we have mentioned in example 24, the empty set is the minimal set, while the sets $\emptyset$, $\{a\}$ and $\{b\}$ are the completely rooted sets. sets. Moreover, the sets $\{a\}$, and $\{b\}$ are the preferred and stable sets.

6.2 Comparison of robust semantics with acceptability of [KMD94]

In [KMD94] the authors present an elegant general acceptability theory for any non-monotonic reasoning framework, which consists of a background logic $\mathcal{L}$ along with a binary attack relation between sets of sentences of $\mathcal{L}$. The general theory is summarized as follows:

Definition 20 ([KMD94]) *Let $\mathcal{T}$ be a theory in a non-monotonic reasoning framework. Then the acceptability relation Acc on $\mathcal{T}$ is specified via the following axioms. For any $T, T_0 \subseteq \mathcal{T}$:*

- $Acc(T, T_0)$ *if* $T \subseteq T_0$
- $Acc(T, T_0)$ *if for any attack T' against T relative to T_0, $\neg Acc(T', T \cup T_0)$.*

The set T is said to be acceptable iff $Acc(T, \emptyset)$.

It is important to notice that acceptability is a binary relation, i.e. that the central notion is "T is acceptable with respect to T_0", where T_0 is regarded as a given choice or context of assumptions. Notice also that the attacks T_0 against T that must be considered are only those that attack the new part of T_0, namely $T \backslash T_0$. This subtraction of T_0 is important as it is needed to ensure that attacks are not against hypotheses in T_0, which we are in fact trying to adopt. In other words, the notion of attack must be relative to a given set of hypotheses (namely T' attacks T relative to T_0) that limits the attack only to those that are against the new hypotheses in T.

It should be noted that the above is a specification rather than a definition. A proper definition which satisfies the specification, in terms of least fixpoints, can be found in [KMD94].

We will adapt definition 20, abstracting away from the background logic $\mathcal{L}$, so that it can be used to define a semantics for any argumentation framework[8]. For our adaptation, we equate $\mathcal{T}$ with the set A of arguments of an argumentation framework $(A, \rightsquigarrow)$. Since the attacking relation deals with subsets of $\mathcal{T}$, we will define relative attack on sets of arguments, but based on the underlying attack relation $\rightsquigarrow$ on arguments.

In view of the above, definition 20 can now be paraphrased as follows: given a context T_0 (i.e. a set of arguments that has been "decided"), another set T can be accepted if any relative attack T' on the new part of T can be shown to be unacceptable, in the extended context $T \cup T_0$.

In light of the above intuition, "relative attack" can be defined as follows[9]:

[8] Notice that, since the background logic might induce some properties on the argumentation framework, where arguments are subsets of $\mathcal{L}$, it is not necessarily true that any argumentation framework can be represented using $\mathcal{L}$.

[9] It should be noted that [KMD94] consider the notion of "relative attack" as a parameter that needs to satisfy some requirements. The relative attack of definition 21 satisfies these requirements.

Definition 21 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework and let $S, T, R \subseteq A$. We write $S \hookrightarrow (T, R)$ iff $S \rightsquigarrow T \backslash R$.*

The following example shows that the attack relation of definition 21 defines an acceptability semantics on argumentation frameworks which is different from the robust semantics.

Example 26 Let $AF = (A, \rightsquigarrow)$ be the argumentation framework shown in figure 8, and let Acc be the acceptability relation defined using the relative attack relation $\hookrightarrow$ defined in definition 21. We have $Acc(\{b\}, \emptyset)$. Note that $\{b\}$ is not acceptable according to the labeling semantics.

Example 26 shows that, considered as a semantics for argumentation frameworks, the acceptability of [KMD94] produces different results from our semantics. It should be noted, however, that this does not indicate a difference in approach to the underlying reasoning framework. For example, in [KMD94] the logic program of example 11 is mapped onto an argumentation framework in such a way that the [KMD94]-acceptable extension $\{\neg q\}$ is the same one as the robust set of the argumentation framework in figure 8.

6.3 Comparison of labelings with status assignments of [Pol94]

In [Pol94], the author introduces a semantics for so-called "inference graphs". That semantics does not attempt to capture such subtleties as those presented, for example, in the robust semantics, but it can be compared to the more general semantics provided by labelings. In section 6.3 we compare labelings of argumentation frameworks and his semantics for inference graphs, given by so-called "status assignments". An **inference graph** is defined in [Pol94] by a triple $(S, T, \rightharpoonup)$. The set S designates nodes which represent **stages of reasoning**, the set T consists of ordered pairs of nodes, or directed edges, which represent **inference links**, and the set $\rightharpoonup$ consists of ordered pairs of nodes which represent **defeat** between nodes. The pair (a, b) is an inference link iff a was inferred (in one step) from a set of nodes, one of which was b. In this case b is said to be an **immediate ancestor** of a. If a was inferred in a series of steps from a set of nodes containing b, then b is referred to as an **ancestor** of a. Although [Pol94] does not state this explicitly, we will assume that every node is inferred in a *finite* number of steps.

Throughout the paper, the author suggests several different definitions for sets of acceptable nodes. In principle 3 of his paper, he suggests his improved version of principles 1 and 2, which is a mapping of the nodes of an inference graph into the set $\{$*"undefeated"*, *"defeated outright"*, *"provisionally defeated"*$\}$:

Definition 22 ([Pol94]) *Let $(S, T, \rightharpoonup)$ be an inference graph.*

1. *D-initial nodes*[10] *are undefeated.*
2. *Self-defeating nodes*[11] *are defeated outright.*

[10] A node is d-initial iff neither it nor any of its inference ancestors are defeated by any nodes.
[11] A node is self-defeating iff some of its inclusive ancestors defeat others.

3. *If μ is not self-defeating, its immediate ancestors are undefeated, and all nodes defeating μ are defeated outright, then μ is undefeated.*

4. *If μ has an immediate ancestor that is defeated outright, or there is an undefeated node that defeats μ, then μ is defeated outright.*

5. *Otherwise, μ is provisionally defeated.*

The following example demonstrates that the semantics presented in principle 3 is different from our acceptable semantics:

Example 27 Consider the inference graph $(S, \emptyset, \rightharpoonup)$ shown in figure 21.

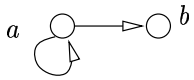


Figure 21: b is undefeated

Clearly, the node a is self-defeating. According to principle 3, therefore, the node a is defeated outright, so the node b is undefeated. As we showed in example 10, the node b is labeled "–", i.e. is defeated in the present framework. Thus, principle 3 provides results which are different from our results.

After having successively rejected principles 1-3, the author makes a final proposal, which is the definition of a status assignment to nodes of an inference graph. He defines this semantics in the following two definitions (taken from definitions 3 and 4 of [Pol94]).

Definition 23 *An assignment σ of "$defeated$" and "$undefeated$" to a subset of the nodes of an inference graph is a* **partial status assignment** *iff:*

1. *σ assigns "$undefeated$" to all d-initial nodes, i.e. all nodes which are neither attacked by any nodes, nor have any inference ancestors which are attacked by any nodes.*

2. *σ assigns "$undefeated$" to a node a iff σ assigns "$undefeated$" to all the immediate ancestors of a and all nodes attacking a are assigned "$defeated$".*

3. *σ assigns "$defeated$" to a node a iff either a has an immediate ancestor that is assigned "$defeated$", or there is a node b that attacks a and is assigned "$undefeated$".*

Definition 24 *σ is a* **status assignment** *iff σ is a partial status assignment and σ is not properly contained in any other partial status assignment.*

[Pol94] states that he sees "no way to recast the present analysis in terms of a defeat relation between arguments (as opposed to nodes, which are argument steps rather than complete arguments)". We shall show that this comment is not valid, by showing that it is possible to represent an inference graph as an argumentation framework, and to determine a semantics for the argumentation framework which corresponds to status

assignments. We first show, in definition 25 and theorem 16, that partial status assignments of an inference graph are equivalent to the completely rooted semantics, which we suggested in definition 19, of an associated argumentation framework. We will then show, in theorem 17, that [Pol94]'s suggested semantics, given by status assignments, is equivalent to the preferred semantics, which was defined in [Dun95] and given in definition 2, of the associated argumentation framework. As can be seen in figure 19, the preferred semantics is a restrictive semantics, in that it does not capture the models suggested by more general semantics such as the completely robust semantics.

Definition 25 *Let $G = (S, T, \rightharpoonup)$ be an inference graph. The* **argumentation framework** $AF = (A, \rightsquigarrow)$ **associated to** G *is defined as follows:*

- $A = S$.
- $\forall a, b \in A$, $a \rightsquigarrow b$ *iff either* $a \rightharpoonup b$ *or* $\exists c \in A$ *such that* c *is an ancestor of* b *and* $a \rightharpoonup c$.

Definition 25 shows how to represent an inference graph as an argumentation framework. Each node in this argumentation framework represents not a stage of reasoning, but an entire argument which includes all of the stages of reasoning that are needed to infer that argument. Thus, the attacks on a node include all the defeat relations on the stages of reasoning that lead to that argument. In theorems 16 and 17 we determine a semantics for this associated argumentation framework, which corresponds to [Pol94]'s status assignments.

Theorem 16 *Let $G = (S, T, \rightharpoonup)$ be an inference graph and $AF = (A, \rightsquigarrow)$ be the argumentation framework associated to G. Let $\sigma : S \rightarrow \{defeated, undefeated, \emptyset\}$ and $l : A \rightarrow 2^{\{+,-\}}$ be total mappings such that*

$$\begin{cases} l(a) = + \text{ iff } \sigma(a) = \text{“}undefeated\text{”} \\ l(a) = - \text{ iff } \sigma(a) = \text{“}defeated\text{”} \\ l(a) = \pm \text{ iff } \sigma(a) = \emptyset. \end{cases}$$

The mapping σ is a partial status assignment iff the mapping l is a rooted complete labeling.

Theorem 17 *Let G, AF, σ and l be as defined in theorem 16. The mapping σ is a status assignment iff the mapping l corresponds to a preferred set of arguments (i.e. iff there is a preferred set S of arguments such that $S = l^+$).*

[Pol94] suggests two benchmark problems which encouraged him to define status assignments as in definition 24. Since we have shown in the previous section that status assignments are just equivalent to certain types of labelings, any problem that can be handled by status assignments can also be handled by labelings. As an example, we will consider the first of the two benchmark problems, which is the lottery paradox.

In [Pol94], the lottery paradox is described as follows: “Suppose you hold one ticket in a fair lottery consisting of one million tickets, and suppose it is known that one and only one ticket will win. Observing that the probability is only 0.000001 of a ticket being drawn given that it is a ticket in the lottery, it seems reasonable to infer defeasibly that your ticket will not win. But by the same reasoning, it will be reasonable to believe,

for each ticket, that it will not win. These conclusions conflict jointly with something else we are warranted in believing, namely, that some ticket will win. Assuming that we cannot be warranted in believing each member of an explicitly contradictory set of propositions, it follows that we are not warranted in believing of each ticket that it will not win."

This situation is easily translated to an argumentation framework with one million nodes i, each of which represents the claim "ticket i is a winning ticket". Each node in the argumentation framework attacks every other node. Labelings handle this argumentation framework naturally, by representing each possible outcome of the lottery. For every node i, there is a complete labeling which assigns $+$ to it and which assigns $-$ to every other node. This labeling describes the fact that i is the winning ticket. There are also 999,999 complete labelings each of which assigns $-$ to i and $+$ to one of the other nodes. These labelings describe the fact the ticket i does not win.

All of these previous labelings are included in the preferred semantics (see figure 19), which is a credulous semantics. They correspond to the possible *outcomes* of the lottery. Before the lottery takes place, someone who wants to decide whether or not to buy a ticket, must refer to a sceptical semantics, which is represented by the remaining complete labelings of the above argumentation framework. One of the labelings assigns $\pm$ to each of the 1,000,000 nodes, and conveys the idea that the given information does not really dictate a decision, so the person can do what she wants. Alternatively, if the person considers buying a particular ticket or group of tickets, she can refer to the complete labeling which assigns $-$ to each of the considered tickets and assigns $\pm$ to all the other tickets. This labeling conveys the idea that she should not buy those tickets, although it does not decide which ticket is the winning ticket.

Thus, labelings provide all the possible views of the lottery paradox. This seems the most reasonable approach, since some people like buying lottery tickets and some do not.

The [Pol94] approach to the lottery paradox is not actually explicitly described in his paper, since he only discusses the lottery paradox in the context of his initial attempts to define an appropriate semantics, and not in the context of his final proposal. However, in light of the equivalence of his semantics and the preferred semantics, it is obvious that status assignments correspond to the possible outcomes of the lottery; i.e. each status assignment corresponds to one of the 1,000,000 complete labelings which assigns $+$ to one of the 1,000,000 nodes and $-$ to all the other nodes. Thus, the other labelings mentioned above have no equivalent in the semantics suggested by [Pol94].

7 Conclusions and directions for further research

We have presented a theory of argumentation that can deal with contradiction within an argumentation framework, and have thus provided a possible solution to a problem posed in [Dun95]. We have shown that our approach unifies several existing semantics of argumentation frameworks and, in addition, produces reasonable and necessary models that are not captured by previous semantics. With the single simple notion of a "(robust) labeling", we have described both the global and the local validity of a set of conclusions. We have shown that the robust semantics captures both credulous and sceptical semantics and, thus, unifies these seemingly incompatible approaches, without resorting to extra operations such as intersection. The latter is called the minimal

semantics in section 4.1. We have shown that the minimal semantics is included in the robust semantics. We have shown that every argumentation framework can be associated to a so-called *meta*-argumentation framework, and that the minimal semantics of the meta-argumentation framework generates the robust semantics of the argumentation framework.

There are some applications for which the various semantics presented in this paper are not sufficient. Recall the medical problem discussed in example 6. As we have explained in that example, it is possible that each of the two types of bacteria is present in the blood, but neither is certain. A credulous semantics would guess at which of the two types of bacteria is present, and thus suggest either of the robust labelings (2) and (3) of figure 7. A sceptical semantics, on the other hand, such as the minimal semantics, would abstain from any decisions and thus suggest the robust labeling (1). However, it seems unreasonable to allow the uncertainty regarding the type of bacteria to prevent the doctor from taking the decision to prescribe the antibiotics since, whichever one of the two types of bacteria is present, the antibiotics is necessary. As we have already stated, therefore, the most reasonable conclusion to this problem is to accept the argument $\{d\}$ as in the robust labeling (4).

For this type of problem it is necessary to define a semantics which pinpoints which acceptable set is appropriate, by taking the intersection of all credulous approaches. Such a semantics accepts those arguments which are true in any credulous approach. This type of reasoning exists in non-monotonic reasoning [Pol94] and logic programming [SZ90]. For argumentation, intersecting credulous sets is suggested in [Vre97]. In order to adapt this idea to the semantics which we have suggested here, we introduce the obligatory semantics as follows:

Definition 26 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. A robust set $S \subseteq A$ of arguments is* **maximal** *if it is not included in any other robust set. The* **obligatory set** *is the intersection of all maximal robust sets.*

The following example demonstrates how to apply definition 26.

Example 28 Consider again the medical problem discussed in example 6. The robust labelings of this argumentation framework are the labelings (1), (2), (3) and (4) of figure 7. (Recall from example 17 that the labeling of figure 13 was not robust.) The maximal robust labelings of this argumentation framework are the labelings (2), (3) and (4) of figure 7. The robust sets are therefore $\emptyset$, $\{a, d\}$, and $\{b, d\}$, so the maximal robust sets are $\{a, d\}$ and $\{b, d\}$. Thus, the obligatory set is $\{d\}$.

Since the robust semantics captures models which are not captured by previous semantics, so does its intersection, as shown by the following example:

Example 29 Consider again the argumentation framework of figure 3. The obligatory set is the set $\{e\}$. This set is not acceptable according to semantics such as [Dun95] and [BDKT97].

Notice that, in example 28, the obligatory set $\{d\}$ corresponds to the labeling (4) of figure 7 and, in example 29, the obligatory set $\{e\}$ corresponds to the labeling (4) of figure 3. It is natural to wonder whether this is always the case, i.e. whether it is always true that the obligatory set corresponds to a labeling. If so, this would mean that the obligatory semantics is a refinement of the acceptable semantics. The following theorem confirms this hypothesis.

Theorem 18 *Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The set of obligatory arguments of AF is completely acceptable.*

This theorem gives the relationship of the obligatory semantics with the acceptable semantics. We are presently examining its relationships with the other semantics presented in this paper, and with other semantics existing in the literature.

Our consistent approach to the argumentation frameworks in figure 11, demonstrated that arguments can sometimes represent groupings of sub-arguments. This means that a framework can be the homomorphic image of a more detailed framework, of which only the meta-properties are known. In some cases, an observer at the meta-level is not interested or even able to know if, for example, some interior loop is even or odd. we would like to determine an equivalence relation on frameworks, and define a semantics which is consistent on any equivalence class of frameworks[12].

Furthermore, having defined our semantics for argumentation frameworks, we are now examining a corresponding dialectic proof theory. Definition 3, because of its local nature, provides a good starting point for such an attempt.

The theory of argumentation which we have presented in this paper, due to its high level of abstraction, can be applied to many branches of science, including logic programming. We have shown, in [Jak95] and [JV96], how to map literals in a logic program onto arguments of an argumentation framework. In [JV97] we are examining the application of the semantics defined in the present paper, to logic programs. Another area in which argumentation frameworks are particularly useful is to model reasoning among electronic agents, a topic that we are currently exploring.

8 Acknowledgments

The first author acknowledges the support of the Foundation for Scientific Research, FWO. The authors are grateful to Donald Nute for useful comments on an early version of this paper. Thanks to the anonymous referees for constructive criticism, and remarks on possible extensions to this paper.

9 Appendix: Proofs

Theorem 1: Every argumentation framework has a complete labeling, and therefore a completely acceptable set.

Proof: Immediate from theorem 5. □

Lemma 1: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework.

- If L is a set of labelings of AF then $\sqcup L$ is a labeling of AF.
- If L is a non-empty set of complete labelings of AF then $\sqcup L$ is a complete labeling of AF.
- If l_1 is a labeling and l_2 is a complete labeling of $AF|_{l_1^0}$ then $l_1 \sqcup l_2$ is a complete labeling.

[12] See also example 11, page 15 for some preliminary thoughts on this topic

Proof: The first two statements of the lemma are straightforward. We prove the third statement by verifying the conditions of definition 3 as follows:

1. Let $a \in A$ be such that $l_1 \sqcup l_2(a) \ni -$. If, on one hand, $a \in l_1^{\emptyset}$ then $\exists b \rightsquigarrow a$ such that $l_2(b) \ni +$. If, on the other hand, $a \in A \backslash l_1^{\emptyset}$ then $\exists b \rightsquigarrow a$ such that $l_1(b) \ni +$. In either case $l_1 \sqcup l_2(b) \ni +$.

2. Let $a \in A$ be such that $l_1 \sqcup l_2(a) \ni +$ and let $b \rightsquigarrow a$. If, on one hand, $a \in A \backslash l_1^{\emptyset}$ then $b \in A \backslash l_1^{\emptyset}$ and $l_1(b) \ni -$, so $l_1 \sqcup l_2(b) \ni -$. If, on the other hand, $a \in l_1^{\emptyset}$ then either $b \in l_1^{\emptyset}$ or $b \in A \backslash l_1^{\emptyset}$. If $b \in l_1^{\emptyset}$ then $l_2(b) \ni -$. If $b \in A \backslash l_1^{\emptyset}$ then we must have $l_1(b) = -$ since, otherwise, $l_1(b) \ni +$ so $l_1(a) \ni -$, which would imply that $a \in A \backslash l_1^{\emptyset}$. Thus, in either case $l_1 \sqcup l_2(b) \ni -$.

3. Let $a, b \in A$ be such that $l_1 \sqcup l_2(a) \ni +$ and let $a \rightsquigarrow b$. If, on one hand, $a \in A \backslash l_1^{\emptyset}$ then $b \in A \backslash l_1^{\emptyset}$ and $l_1(b) \ni -$, so $l_1 \sqcup l_2(b) \ni -$. If, on the other hand, $a \in l_1^{\emptyset}$ then either $b \in l_1^{\emptyset}$ or $b \in A \backslash l_1^{\emptyset}$. In the former case $l_2(b) \ni -$. In the latter case we must have $l_1(b) = -$ since, otherwise, $l_1(b) \ni +$ so $l_1(a) \ni -$, which would imply that $a \in A \backslash l_1^{\emptyset}$. In either case $l_1 \sqcup l_2(b) \ni -$.

$l_1 \sqcup l_2$ is complete by definition. □

Theorem 2: Let S be an acceptable set of an argumentation framework AF. There is a completely acceptable set T such that $S \subseteq T$.

Proof: Let l_1 be a labeling which corresponds to S. Let l_2 be a complete labeling of $AF|_{l_1^{\emptyset}}$. (By theorem 1, every argumentation framework has a complete labeling, so such a complete labeling l_2 exists.) By lemma 1, $l_1 \sqcup l_2$ is a complete labeling. Clearly, if T is the completely acceptable set which corresponds to $l_1 \sqcup l_2$ then $S \subseteq T$. □

Lemma 2: Let S be a defendable set of arguments. Let T and T' be supporting defences of S. Then $T \cup T'$ is a supporting defence of S.

Proof: Straightforward. □

Theorem 3: Let S be an acceptable set of an argumentation framework $AF = \{A, \rightsquigarrow\}$, and let l be a labeling that corresponds to S. The set S is defendable and $l^{\pm}$ is a supporting defence of S.

Proof: Since S corresponds to l we have $S = l^{+}$. It is clear that S is consistent; indeed, assume the contrary, i.e. that $\exists a, b \in S$ such that $a \rightsquigarrow b$. By condition 3 of definition 3, $- \in l(b)$, which contradicts the fact that $b \in l^{+}$. Let $T = l^{\pm}$. We show that T is a supporting defence of S, by verifying the conditions of definition 7:

1. Let $a \in A$ be such that $a \rightsquigarrow b \in S \cup T$. Since $l(b) \ni +$, condition 2 of definition 3 implies that $l(a) \ni -$. Thus, by condition 1 of definition 3, $S \cup T \rightsquigarrow a$.

2. By condition 2 of definition 3, $S \not\rightsquigarrow T$.

3. By condition 3 of definition 3, $T \not\rightsquigarrow S$.

4. $S \cap T = \emptyset$, by definition of l.

□

Theorem 4: Let S be a maximal defendable set of an argumentation framework $AF = (A, \rightsquigarrow)$ (i.e. S is defendable and there is no defendable set S' such that $S \subset S'$) and

let T be the maximal supporting defence of S. The set S is completely acceptable and corresponds to a complete labeling l such that $l^{\pm} = T$.

Proof: Defi ne a mapping as follows:

- $\forall a \in S$ let $l(a) = +$.
- $\forall a \in T$ let $l(a) = \pm$.
- $\forall a \in A \backslash (S \cup T)$ let $l(a) = -$.

We show that l is a complete labeling by verifying the conditions of defi nition 3:

1. Let a be such that $- \in l(a)$. We must show that $\exists b \rightsquigarrow a$ such that $+ \in l(b)$. Notice that either $a \in T$ or $a \in A \backslash (S \cup T)$. Suppose fi rst that $a \in T$. We show that if $T \not\rightsquigarrow a$ then $T \backslash \{a\}$ is a supporting defence of $S \cup \{a\}$, which contradicts the maximality of S. The conditions of defi nition 7 are verifi ed as follows:

   (a) Let $b \in A$ be such that $b \rightsquigarrow (T \backslash \{a\}) \cup (S \cup \{a\}) = S \cup T$. Since T is a supporting defence of S, we have $S \cup T \rightsquigarrow b$.

   (b) $S \cup \{a\} \rightsquigarrow T \backslash \{a\}$ would imply either that $S \rightsquigarrow T \backslash \{a\}$ or that $\{a\} \rightsquigarrow T \backslash \{a\}$. The former is impossible, since T is a supporting defence of S. The latter implies that $S \cup T \rightsquigarrow a$. Since $T \not\rightsquigarrow a$, we have $S \rightsquigarrow a$. Since $a \in T$, this means that $S \rightsquigarrow T$, which is impossible.

   (c) $T \backslash \{a\} \rightsquigarrow S \cup \{a\}$ is impossible since $T \not\rightsquigarrow S$ and $T \not\rightsquigarrow a$.

   (d) $T \backslash \{a\} \cap (S \cup \{a\}) = \emptyset$, since $T \cap S = \emptyset$.

   Suppose now that $a \in A \backslash (S \cup T)$. We need to show that $S \cup T \rightsquigarrow a$. Suppose, on the contrary, that $S \cup T \not\rightsquigarrow a$. By condition 1 of defi nition 7, then, $a \not\rightsquigarrow S \cup T$. We fi rst show that there must be an argument b such that $b \rightsquigarrow a$ and $S \cup T \not\rightsquigarrow b$. We show that if this is not the case then T is a supporting defence of $S \cup \{a\}$, which contradicts the maximality of S. Indeed, in this case the conditions of defi nition 7 are verifi ed as follows:

   (a) Let $d \in A$ be such that $d \rightsquigarrow S \cup \{a\} \cup T$. Then either $d \rightsquigarrow a$ or $d \rightsquigarrow S \cup T$. In the former case, since we have supposed that $\nexists b \rightsquigarrow a$ such that $S \cup T \not\rightsquigarrow b$, we must have $S \cup T \rightsquigarrow d$, so the condition is satisfi ed. In the latter case, since T is a supporting defence of S, this also gives $S \cup T \rightsquigarrow d$.

   (b) If $S \cup \{a\} \rightsquigarrow T$ then either $S \rightsquigarrow T$ or $\{a\} \rightsquigarrow T$. The former cannot hold since T is a supporting defence of S. The latter cannot hold since we have supposed that $a \not\rightsquigarrow S \cup T$.

   (c) $T \rightsquigarrow S \cup \{a\}$ would either imply that $T \rightsquigarrow S$ or $T \rightsquigarrow \{a\}$, both of which are impossible.

   (d) $(S \cup \{a\}) \cap T = \emptyset$, since $S \cap T = \emptyset$ and we have taken $a \in A \backslash (S \cup T)$.

   Thus, there is an argument b such that $b \rightsquigarrow a$ and $S \cup T \not\rightsquigarrow b$. Note that, since T is a supporting defence of S, this means that $b \not\rightsquigarrow S \cup T$. Since $S \cup T \not\rightsquigarrow a$, we must have $b \in A \backslash (S \cup T)$. By the same reasoning, there is an argument c such

that $c \rightsquigarrow b$, $S \cup T \not\rightsquigarrow c$, $c \not\rightsquigarrow S \cup T$, and $c \in A \backslash (S \cup T)$. This reasoning can be repeated, and we consider therefore the following function:

$$\phi : 2^{A \backslash (S \cup T)} \rightarrow 2^{A \backslash (S \cup T)}$$

$$\phi(X) = \{a\} \cup \{d \mid d \rightsquigarrow X \text{ and } S \cup T \not\rightsquigarrow d\}$$

This function is clearly monotonic with respect to set inclusion so, as shown in [Tar55], ϕ has a least fixpoint Φ. It is straightforward to show that $T \cup \Phi$ is a supporting defence of S, which contradicts the maximality of T.

2. Let a be such that $+ \in l(a)$. We must show that $\forall b \rightsquigarrow a, - \in l(b)$. Notice that either $a \in S$ or $a \in T$. Suppose first that $a \in S$. Since S is consistent, $b \notin S$. Since T is a supporting defence of S, $b \notin T$. So $b \in A \backslash (S \cup T)$ and $l(b) = -$. Now suppose that $a \in T$. Since T is a supporting defence of S, $b \notin S$. So $b \in T$ or $b \in A \backslash (S \cup T)$. In both cases $l(b) \ni -$.

3. Let a be such that $+ \in l(a)$. We must show that $\forall b \in A$ if $a \rightsquigarrow b$ then $- \in l(b)$. Again, either $a \in S$ or $a \in T$. If $a \in S$ then, by the consistency of S, $b \notin S$. So $l(b) \ni -$. If $a \in T$ then, since T is a supporting defence of S, $b \notin S$. So $l(b) \ni -$.

Thus, l is a labeling. By definition of the function l, it is complete. □

Lemma 3: Let $(A, \rightsquigarrow)$ be an argumentation framework. The fixpoints S and T, defined in definition 9, are disjoint.

Proof: We show, by induction on α, that $\forall \beta, \gamma$ such that $\beta + \gamma = \alpha$, $S_\beta \cap T_\gamma = \emptyset$. When $\alpha = 0$, the statement trivially holds. Suppose, for the induction hypothesis, that $\forall \alpha' < \alpha$ the statement holds. Suppose that the statement does not hold for α. This means that $\exists \beta \exists \gamma$ such that $\beta + \gamma = \alpha$ and $S_\beta \cap T_\gamma \neq \emptyset$. Let $a \in S_\beta \cap T_\gamma$. We first consider the case in which both β and γ are successor ordinals. Since $a \in S_\beta$, $\forall b \rightsquigarrow a, b \in T_{\beta-1}$. Since $a \in T_\gamma$, $S_\gamma \rightsquigarrow a$. Thus, $S_\gamma \cap T_{\beta-1} \neq \emptyset$. Since $\gamma + \beta - 1 < \beta + \gamma$, this contradicts the induction hypothesis. We next consider the case in which at least one of the ordinals β, γ is a limit ordinal. E.g., suppose that β is a limit ordinal. By definition of S_β, there is a successor ordinal $\beta' < \beta$ such that $a \in S_{\beta'}$. Since $\beta' + \gamma < \beta + \gamma$, this contradicts the induction hypothesis. □

Theorem 5: The mapping l_{min} defined in definition 10 is a complete labeling.

Proof: Straightforward. □

Theorem 6: Any labeling of an argumentation framework $AF = (A, \rightsquigarrow)$ is a refinement of the minimal complete labeling of AF. In particular, any argument $a \in A$ which is accepted in the minimal complete labeling (i.e. $l_{min} = +$) is accepted in any complete labeling. Similarly, any argument $b \in A$ which is rejected in the minimal complete labeling (i.e. $l_{min} = -$) is rejected in any complete labeling.

Proof: Let l be a labeling of AF. We first show that there is a complete labeling $\boldsymbol{l}$ such that l is a refinement of $\boldsymbol{l}$. By theorem 2, there is complete labeling l_c of $AF|_{l^\emptyset}$. Clearly, $l' = l \sqcup l_c$ is a complete labeling of AF, and l is a refinement of $\boldsymbol{l}$.

We now show that l' is a refinement of the minimal complete labeling l_{min}. By the transitivity of $\sqsubseteq$, this will imply that l is a refinement of l_{min}.

Suppose that l' is not a refinement of l_{min}. This means that there is an argument a such that either we have

$$l_{min}(a) = + \text{ and } l'(a) \ni -$$

or we have

$$l_{min}(a) = - \text{ and } l'(a) \ni +.$$

In the first case it can be shown that there is an ordinal α such that $a \in S_\alpha \cap W$ (see definitions 9 and 10), and in the second case there is an ordinal α such that $a \in T_\alpha \cap Y$, where

$$W = \{a \in l^+_{min} \mid l'(a) \ni -\}$$

and

$$Y = \{a \in l^-_{min} \mid l'(a) \ni +\}.$$

Let α be the least ordinal such that $(S_\alpha \cap W) \cup (T_\alpha \cap Y) \neq \emptyset$. Notice that α cannot be a limit ordinal because otherwise, by definition of S_α and T_α, $\exists \beta < \alpha$ such that $(S_\beta \cap W) \cup (T_\beta \cap Y) \neq \emptyset$. This would contradict the minimality of α, so α is a successor ordinal. Since $S_\alpha \cap W$ and $T_\alpha \cap Y$ are disjoint we have either $a \in S_\alpha \cap W$ or $a \in T_\alpha \cap Y$.

In the former case, since $l'(a) \ni -$ we have, by condition 1 of definition 3, that $\exists b \rightsquigarrow a$ such that $l'(b) \ni +$. Since $l_{min}(a) = +$ we must have $l_{min}(b) = -$; therefore, $b \in Y$. In addition, we must have $a \in S_\alpha$ so, by definition of S_α, we have $b \in T_{\alpha-1}$. This implies that $b \in T_{\alpha-1} \cap Y$, which contradicts the minimality of α.

In the latter case $a \in T_\alpha$ so, by definition of T_α, there must be an argument b such that $b \rightsquigarrow a$ and $b \in S_{\alpha-1}$. Since $l_{min}(a) = -$ and $l'(a) \ni +$, we have $l_{min}(b) = +$ and $l'(b) \ni -$. This implies that $b \in S_{\alpha-1} \cap W$, which contradicts the minimality of α. □

Lemma 4: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. Let L be a set of refinements of a labeling l of AF. The labeling $\sqcup L$ is a refinement of l.

Proof: Straightforward. □

Corollary 2 Let $\mathcal{L}$ be the set of all labelings of an argumentation framework AF. $\mathcal{L}_\sqsubseteq$ is a complete lattice, i.e. $\sqsubseteq$ is a partial order and for any $S \subseteq \mathcal{L}$, there are labelings $lub(S)$ and $glb(S)$. Furthermore, $lub(S) = \sqcup S$, and the top element of $\mathcal{L}_\sqsubseteq$ is the minimal complete labeling of AF.

Proof: It is straightforward to show that $\sqsubseteq$ is a partial order. By lemma 1, $\sqcup S$ is a labeling. Every element in S is clearly a refinement of $\sqcup S$, so $\sqcup S$ is indeed an upper bound of S. By lemma 4, $\sqcup S$ is a refinement of any upper bound of S, so $\sqcup S$ is the least upper bound of S. Thus, $\mathcal{L}_\sqsubseteq$ is a complete semi-lattice. It is a well-known fact that any complete semi-lattice is a complete lattice, so there is a labeling $glb(S)$. Finally, by theorem 6 any $l \in \mathcal{L}$ is a refinement of the minimal complete labeling l_{min} of AF so l_{min} is the top element. □

Lemma 5: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The function N, defined in definition 16, has a least fixpoint.

Proof: This results from lemma 6. □

Theorem 7: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The set of robust restricted labelings satisfies specification 2, i.e. if X is the set of robust restricted labelings then $X = \{l \in A^* \mid \forall l' \rightsquigarrow^* l, l' \notin X \wedge \exists l'' \rightsquigarrow l' \text{ such that } l'' \in X\}$.

Proof: Notice first that a set X of restricted labelings satisfies the specification iff it is a fixpoint of the function N. Indeed,

$$\begin{aligned}
& X \text{ is a fixpoint of } N \\
\iff\ & X = N(X) \\
\iff\ & l \in X \iff l \in N(X) \\
\iff\ & l \in X \iff \forall l' \rightsquigarrow l, l' \notin X \wedge \exists l'' \rightsquigarrow^* l' \text{ such that } l'' \in X \\
\iff\ & X \text{ satisfies the specification}
\end{aligned}$$

Since the set of robust restricted labelings is, by definition, the least fixpoint of N, it is a fixpoint of N, so it satisfies the specification. □

Lemma 6: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework and let $AF^* = (A^*, \rightsquigarrow^*)$ be its meta-argumentation framework. The set S^* is the least fixpoint of the function N, and is therefore the set of robust restricted labelings.

Proof: We first show that S^* is a fixpoint of N, i.e. that $S^* = N(S^*)$. This is the case iff $S^* = \{l \in A^* \mid \forall l' \rightsquigarrow^* l, l' \notin S^* \wedge \exists l'' \rightsquigarrow^* l' \text{ such that } l'' \in S^*\}$.

To show that $S^* \subseteq N(S^*)$, let $l \in S^*$. According to definition 18, there is a least ordinal α such that $l \in S^*_\alpha$. Since α is least, it is a successor ordinal. By definition of S^*_α, $\forall l' \rightsquigarrow^* l, l' \in T^*_{\alpha-1}$ so $l' \in T^*$. By lemma 3, S^* and T^* are disjoint, so $l' \notin S^*$. Since $l' \in T^*_{\alpha-1}$, $S^*_{\alpha-1} \rightsquigarrow^* l'$, by definition of $T^*_{\alpha-1}$. Therefore, $\exists l'' \rightsquigarrow^* l'$ such that $l'' \in S^*_{\alpha-1}$. Since $S^*_{\alpha-1} \subseteq S^*$, $l'' \in S^*$. We have thus shown that $l \in N(S^*)$.

To show that $N(S^*) \subseteq S^*$, let $l \in N(S^*)$. This implies that $\forall l' \rightsquigarrow^* l, \exists l'' \rightsquigarrow^* l'$ such that $l'' \in S^*$. Since $S^* = \bigcup_{\alpha \in On} S^*_\alpha$, this means that $\forall l' \rightsquigarrow^* l$ there is a least ordinal α such that $S_\alpha \rightsquigarrow^* l'$. Since α is least, it is a successor ordinal. By definition of T^*_α, then, $\forall l' \rightsquigarrow^* l, l' \in T^*_\alpha$. Therefore, by definition of $S^*_{\alpha+1}$, $l \in S^*_{\alpha+1}$, so $l \in S^*$.

To show that S^* is the least fixpoint of N, we need to show that for any fixpoint X of N, $S \subseteq X$. Let X be a fixpoint of N, i.e. $X = N(X)$. We show by transfinite induction on α that $S^*_\alpha \subseteq X$ and $T^*_\alpha \cap X = \emptyset$. For $\alpha = 0$ the statement trivially holds. For the induction hypothesis, suppose that the statement is true for all $\beta < \alpha$.

We first consider the case in which α is a successor ordinal. If the statement is not true then there is a restricted labeling l such that either $l \in S^*_\alpha \backslash X$ or $l \in T^*_\alpha \cap X$. Suppose first that $l \in S^*_\alpha \backslash X$. Since $l \notin X = N(X)$, $\exists l' \rightsquigarrow^* l$ such that either $l' \in X$ or $\forall l'' \rightsquigarrow^* l', l'' \notin X$. By definition of S^*_α, $l' \in T^*_{\alpha-1}$. Therefore, in the former case, $l' \in T_{\alpha-1} \cap X$, which contradicts the induction hypothesis. In the latter case $\forall l'' \rightsquigarrow^* l', l'' \notin X$. By definition of $T^*_{\alpha-1}$, $S^*_{\alpha-1} \rightsquigarrow^* l'$. Therefore, $\exists l'' \rightsquigarrow l'$ such that $l'' \in S^*_{\alpha-1}$. Since $l'' \notin X$, $l'' \in S^*_{\alpha-1} \backslash X$, which contradicts the induction hypothesis.

Suppose now that $l \in T^*_\alpha \cap X$. By definition of T^*_α, $S^*_\alpha \rightsquigarrow^* l$, so $\exists l' \rightsquigarrow^* l$ such that $l' \in S^*_\alpha$. By definition of S^*_α, $\forall l'' \rightsquigarrow^* l', l'' \in T^*_{\alpha-1}$. By the induction hypothesis, then, $\forall l'' \rightsquigarrow^* l', l'' \notin X$. Thus, it is not true that $\forall l' \rightsquigarrow^* l, \exists l'' \rightsquigarrow^* l'$ such that $l'' \in X$, so $l \notin N(X) = X$. Therefore, $l \notin T^*_\alpha \cap X$, which is a contradiction.

Next, let α be a limit ordinal. If the statement is not true then $\exists l \in A^*$ such that either $l \in S^*_\alpha \backslash X$ or $l \in T^*_\alpha \cap X$. Since $S^*_\alpha = \bigcup_{\beta<\alpha} S^*_\beta$ and $T^*_\alpha = \bigcup_{\beta<\alpha} T^*_\beta$, this means that there is an ordinal $\beta < \alpha$ such that either $l \in S^*_\beta \backslash X$ or $l \in T^*_\beta \cap X$. This contradicts the induction hypothesis. □

Theorem 8: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let l^*_{min} be the minimal complete labeling of $AF^* = (A^*, \rightsquigarrow^*)$. The restricted labelings that are

labeled $+$ are the robust restricted labelings of AF, i.e.

$$l^*_{min}{}^{+} = \{l \in A^* \mid l \text{ is robust } \}.$$

Proof: This results from lemma 6. □

Corollary 3: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let l^*_{min} be the minimal complete labeling of AF^*. The set of robust labelings of AF is the set $\{l \in l^*_{min}{}^{+} \mid$ the domain of l is $A\}$.

Proof: This is a result of theorem 8. □

Lemma 7: Let l be a rooted labeling of an argumentation framework $AF = (A, \rightsquigarrow)$. Let l' be the minimal complete labeling of $AF|_{l^\emptyset}$. The complete labeling $l \sqcup l'$ is rooted.

Proof: Notice first that, by lemma 1, $l \sqcup l'$ is indeed a labeling. Clearly, it is complete. By theorem 11, the minimal complete labeling is rooted. Thus, since l and l' are each rooted, it is straightforward to show that $l \sqcup l'$ is rooted. □

Theorem 9: Let S be a rooted set of an argumentation framework $AF = (A, \rightsquigarrow)$. There is a completely rooted set T such that $S \subseteq T$.

Proof: Let l be a rooted labeling which corresponds to S and let l' be as defined in lemma 7. By lemma 7, $l \sqcup l'$ is a rooted complete labeling. Clearly, if T is the completely rooted set which corresponds to $l \sqcup l'$, then $S \subseteq T$. □

Lemma 8 *Let l be a labeling of an argumentation framework $AF = (A, \rightsquigarrow)$, and l' be an extension of l. Then $l \dagger l'$ satisfies conditions 2 and 3 of definition 3.*

Proof: To show that condition 2 holds, let a be such that $l \dagger l'(a) \ni +$ and let $b \rightsquigarrow a$. We must either have $l(a) = +$ or $l(a) = \pm$. In the former case $l(b) = -$ so $l \dagger l'(b) = -$. In the latter case, either $l(b) = -$ or $l(b) = \pm$. If $l(b) = -$ then $l \dagger l'(b) = -$. If $l(b) = \pm$ then, since $l'(a) \ni +$, we have $l'(b) \ni -$, which yields $l \dagger l'(b) \ni -$.

To show that condition 3 holds, let a be such that $l \dagger l'(a) \ni +$, and let b be such that $a \rightsquigarrow b$. Then either $l(a) = +$ or $l(a) = \pm$. If $l(a) = +$ then $l(b) = - = l \dagger l'(b)$ and the condition holds. If, on the other hand, $l(a) = \pm$ then l' is defined on a and $l'(a) \ni +$. Since $l(a) \ni +$ we must have $l(b) \ni -$, so we must have either $l(b) = -$ or $l(b) = \pm$. In the former case $l \dagger l'(b) = l(b) = -$ and the condition holds. In the latter case l' is defined on b and $l \dagger l'(b) = l'(b) \ni -$. □

Theorem 10: All rooted labelings are robust.

Proof: To show that rooted labelings are in the least fixpoint of the function N, it suffices to show that rooted labelings have no incompatible extensions. Let l be a rooted labeling. By lemma 8, it suffices to show that for any extension of l, $l \dagger l'$ satisfies condition 1 of definition 3. Let $a \in A$ be such that $l \dagger l'(a) \ni -$. If $l \dagger l'(a) = -$ then either $l(a) = -$ or $l'(a) = -$. In the former case, since l is rooted, $\exists b \rightsquigarrow a$ such that $l(b) = + = l \dagger l'(b)$. In the latter case, since l' is a complete labeling, $\exists b \rightsquigarrow a$ such that $l'(b) = + = l \dagger l'(b)$. If $l \dagger l'(a) = \pm$ then $l(a) = \pm$ and l' is defined on a, so $l'(a) = \pm$; therefore $\exists b \rightsquigarrow a$ such that $l'(b) = \pm = l \dagger l'(b)$. □

Theorem 11: The minimal complete labeling is rooted.

Proof: Straightforward. □

Corollary 6: The minimal set is completely robust.

Proof: This is a result of theorems 10 and 11. □

Theorem 14: A set S of arguments in an argumentation framework AF is a stable set iff there is a complete labeling l which corresponds to S such that $l^{\pm} = \emptyset$.

Proof: Let l be a complete labeling of $AF = (A, \rightsquigarrow)$ such that $l^{\pm}$ is empty. Then $A \backslash S$ is just the set l^{-}. By condition 1) of definition 3, every argument in this set is attacked by S. In addition, by condition 2) of definition 3, S is consistent, so S is stable.

Suppose now that S is stable. Define the mapping l as follows: let $l(a) = +$ if $a \in S$, and $l(a) = -$ if $a \notin S$. It is straightforward to show that l is a complete labeling. □

Theorem 15: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework, and let $S \subseteq A$ be a set of arguments.

1. S is rooted iff S is admissible.
2. S is maximal completely rooted (i.e. S is completely rooted and there is no completely rooted set T such that $S \subset T$) iff S is preferred.

Proof:

1. Suppose S is rooted. Let l be a rooted labeling of AF such that $S = l^{+}$. By condition 3 of definition 3, l^{+} is consistent. Let a be an argument in AF such that $a \rightsquigarrow S$. By conditions 2 and 3 of definition 3, $l(a) = -$. Since l is rooted, $S \rightsquigarrow a$.

   Now suppose S is admissible. Define the following mapping l:

   - $\forall a \in S$ let $l(a) \ni +$.
   - $\forall a \in A$ such that $S \rightsquigarrow a$ let $l(a) \ni -$.

   We show that l is a labeling by verifying conditions 1-3 of definition 3:

   (a) Let a be such that $- \in l(a)$. We must show that $\exists b \rightsquigarrow a$ such that $+ \in l(b)$. Notice that, since S is consistent, there are no arguments labeled $\pm$; therefore, $l(a) = -$. By definition of l, $l^{+} = S \rightsquigarrow a$.

   (b) Let a and b be such that $+ \in l(a)$ and $b \rightsquigarrow a$. Notice that $l(a) = +$, so $a \in S$. Since S is admissible, $S \rightsquigarrow a$. Thus, $l(b) = -$.

   (c) Let a and b be such that $+ \in l(a)$ and $a \rightsquigarrow b$. Since $l^{+} = S \rightsquigarrow b$, we have $l(b) = -$.

2. Let S be a preferred set of AF. Define the following mapping l:

   - $\forall a \in S$ let $l(a) = +$.
   - $\forall a \in A$ such that $S \rightsquigarrow a$ let $l(a) = -$.
   - For any remaining argument $a \in A$ let $l(a) = \pm$.

   We show that l is a labeling by verifying the conditions of definition 3:

(a) Let a be such that $- \in l(a)$. We must show that $\exists b \rightsquigarrow a$ such that $+ \in l(b)$. If $l(a) = -$ then, by definition of l, $I^+ \rightsquigarrow a$. If, on the other hand, $l(a) = \pm$ then, since S is maximal, we have either $S \rightsquigarrow a$ or $a \rightsquigarrow S$ or $\exists b \rightsquigarrow a$ such that $S \cup \{a\} \not\rightsquigarrow b$. In the first case $I^+ \rightsquigarrow a$. In the second case, since S is admissible, $S \rightsquigarrow a$. In the third case either $l(b) = +$ or $l(b) = \pm$, so condition 1 of definition 3 is satisfied.

(b) Let a and b be such that $+ \in l(a)$ and $b \rightsquigarrow a$. If $l(a) = +$ then, since S defends itself, $S \rightsquigarrow b$ so, by definition of l, $l(b) = -$. If, on the other hand, $l(a) = \pm$ then we must have $l(b) \ni -$ since, otherwise, we would have $l(b) = +$ and thus $l(a) = -$.

(c) Let a and b be such that $+ \in l(a)$ and $a \rightsquigarrow b$. If $l(a) = +$ then, by definition of l, $l(b) = -$. If, on the other hand, $l(a) = \pm$ then we must have $l(b) \ni -$ because otherwise $l(b) = +$ so we must have $b \in S$. However, since S defends itself, this would imply that $S \rightsquigarrow a$, so $l(b) = -$, which is a contradiction.

Thus, l is a labeling. Trivially, l is complete.

To show that l is rooted, let a be such that $l(a) = -$. By definition of l, $S \rightsquigarrow a$, so $\exists b \rightsquigarrow a$ with $l(b) = +$.

Thus, S is completely rooted. By definition, S is maximal admissible so, by part 1 of the theorem, S is maximal rooted. Since all completely rooted sets are rooted, S must be maximal completely rooted. Indeed, if there is a completely rooted set T such that $S \subset T$ then, since T is rooted, S would not be maximal rooted, which is a contradiction.

Now let S be maximal completely rooted. By part 1 of the theorem, S is admissible. To show that S is maximal admissible suppose, on the contrary, that there is an admissible set $T \supset S$. By part 1 of the theorem T is rooted so, by theorem 9, there is a completely rooted set $R \supseteq T$. Since $S \subset R$, S is not maximal completely rooted, which is a contradiction.

□

Theorem 16: Let $G = (S, T, \rightharpoonup)$ be an inference graph and $AF = (A, \rightsquigarrow)$ be the argumentation framework associated to G. Let $\sigma : S \rightarrow \{defeated, undefeated, \emptyset\}$ and $l : A \rightarrow 2^{\{+,-\}}$ be total mappings such that $\forall a \in A$,

$$\begin{cases} l(a) = + \text{ iff } \sigma(a) = \text{“}undefeated\text{”} \\ l(a) = - \text{ iff } \sigma(a) = \text{“}defeated\text{”} \\ l(a) = \pm \text{ iff } \sigma(a) = \emptyset. \end{cases}$$

The mapping σ is a partial status assignment iff the mapping l is a rooted complete labeling.

Proof: Let l be a rooted complete labeling. We verify the 3 conditions of definition 23 as follows:

1. Let a be a d-initial node. Clearly, there are no nodes that attack a in the associated argumentation framework. By condition 1 of definition 3, $- \notin l(a)$. Thus, $l(a) = +$, so a is undefeated.

2. To show the first implication, let $a \in S$ be a node such that $\sigma(a) =$ *"undefeated"*. Let $b \in S$ be an immediate ancestor of a. Since $\sigma(a) =$ *"undefeated"*, we have $l(a) = +$ so $\forall c \rightsquigarrow a$ we have $l(c) = -$. Therefore, $\forall d \rightharpoonup b, l(d) = -$, so $l(b) = +$, which implies that $\sigma(b) =$ *"undefeated"*.

   Now let $b \in S$ be such that $b \rightharpoonup a$. Notice that $b \rightsquigarrow a$. Since $l(a) = +$ we have $l(b) = -$, so $\sigma(b) =$ *"defeated"*.

   To prove the second implication, let $a \in S$ be such that for any immediate ancestor c of a, $\sigma(c) =$ *"undefeated"*, and such that $\forall b \rightharpoonup a, \sigma(b) =$ *"defeated"*. To show that $\sigma(a) =$ *"undefeated"* we need to show that $l(a) = +$, so it suffices to show that $\forall d \rightsquigarrow a, l(d) = -$. Since $d \rightsquigarrow a$, either $d \rightharpoonup a$ or there is an ancestor e of a such that $d \rightharpoonup e$. If $d \rightharpoonup a$ then $\sigma(d) =$ *"defeated"* so $l(d) = -$. If, on the other hand, there is an ancestor e of a such that $d \rightharpoonup e$ then there is an immediate ancestor f of a such that e is an ancestor of f. Since f is an immediate ancestor of a we have $\sigma(f) =$ *"undefeated"* so $l(f) = +$. Since $d \rightharpoonup e$ we have $d \rightsquigarrow f$, so $l(d) = -$.

3. To show the first implication, let $a \in S$ be a node such that $\sigma(a) =$ *"defeated"*. This means that $l(a) = -$ so, since l is rooted, $\exists b \rightsquigarrow a$ such that $l(b) = +$, which means that $\sigma(b) =$ *"undefeated"*. Since $b \rightsquigarrow a$, either $b \rightharpoonup a$ (which is one of the cases described in condition 3), or there is an ancestor c of a such that $b \rightharpoonup c$. In this case, $b \rightsquigarrow c$ so $l(c) = -$ and $\sigma(c) =$ *"defeated"*. Since c is an ancestor of a, it is either an immediate ancestor of a (which is one of the cases described in condition 3), or there is an immediate ancestor d of a such that c is an ancestor of d. Since $b \rightsquigarrow c$ we have $b \rightsquigarrow d$. Therefore, since $l(b) = +$, we have $l(d) = -$ so $\sigma(d) =$ *"defeated"*. (This is again one of the cases described in condition 3).

   To show the second implication, let $a \in S$ be such that either a has an immediate ancestor b such that $\sigma(b) =$ *"defeated"*, or $\exists c \rightharpoonup a$ such that $\sigma(c) =$ *"undefeated"*. To show that $\sigma(a) =$ *"defeated"*, we need to show that $l(a) = -$. Suppose that we are in the first case described in condition 3, i.e. that a has an immediate ancestor b such that $\sigma(b) =$ *"defeated"*. Then $l(b) = -$ so, since l is rooted, $\exists d \rightsquigarrow b$ such that $l(d) = +$. Since $d \rightsquigarrow b$ we have $d \rightsquigarrow a$, so $l(a) = -$. Now suppose that we are in the second case, i.e. $\exists c \rightharpoonup A$ such that $\sigma(c) =$ *"undefeated"*. This implies that $l(c) = +$ and $c \rightsquigarrow a$, so $l(a) = -$.

Now let σ be a partial status assignment. We verify the conditions of definition 3 as follows:

1. Let $a \in A$ be such that $- \in l(a)$. This means that either $\sigma(a) =$ *"defeated"* or $\sigma(a) = \emptyset$. In the former case, either a has an immediate ancestor b that is assigned *"defeated"*, or there is a node $d \rightharpoonup a$ such that $\sigma(d) =$ *"undefeated"*. If $d \rightharpoonup a$ and $\sigma(d) =$ *"undefeated"* then $d \rightsquigarrow a$ and $l(d) = +$, so the condition is satisfied. If a has an immediate ancestor b that is assigned *"defeated"* then, since a is inferred in a finite number of steps, it has an ancestor c which is attacked by a node d such that $\sigma(d) =$ *"undefeated"*. Thus, $l(d) = +$ and $d \rightsquigarrow a$, so a satisfies the first condition of definition 3. In the latter case, since σ does not assign *"undefeated"* to a, either a has an immediate ancestor b such that $\sigma(b) \neq$ *"undefeated"* or there is a node $c \rightharpoonup a$ such that $\sigma(c) \neq$ *"defeated"*. If $c \rightharpoonup a$ and $\sigma(c) \neq$ *"defeated"* then $c \rightsquigarrow a$ and $l(c) \ni +$, so the condition is

satisfied. If a has an immediate ancestor b such that $\sigma(b) \neq$ *"undefeated"* then, since a is inferred in a finite number of steps, it has an ancestor d such that $\exists e \rightharpoonup d$ with $\sigma(e) \neq$ *"defeated"*. Thus, $e \rightsquigarrow a$ and $l(e) \ni +$.

2. Let $a \in A$ be such that $+ \in l(a)$, and let $b \in A$ be such that $a \rightsquigarrow b$. Since $+ \in l(a)$, either $\sigma(a) =$ *"undefeated"* or $\sigma(a) = \emptyset$. Suppose first that $\sigma(a) =$ *"undefeated"*. Since $a \rightsquigarrow b$, either $a \rightharpoonup b$ or there is an ancestor c of b such that $a \rightharpoonup c$. In the former case, $\sigma(b) =$ *"defeated"*. In the latter case $\sigma(c) =$ *"defeated"* so $\sigma(b) =$ *"defeated"*. Thus, in both cases $l(b) = -$, so condition 2 of definition 3 is satisfied. Suppose now that $\sigma(a) = \emptyset$. This means that $l(a) = \pm$. Again, since $a \rightsquigarrow b$, either $a \rightharpoonup b$ or there is an ancestor c of b such that $a \rightharpoonup c$. In the former case, since $\sigma(a) \neq$ *"defeated"*, we cannot have $\sigma(b) =$ *"undefeated"*, so $l(b) \ni -$. In the latter case, since $\sigma(a) \neq$ *"defeated"*, we cannot have $\sigma(c) =$ *"undefeated"*. Therefore, we cannot have $\sigma(b) =$ *"undefeated"*. Indeed, if $\sigma(b) =$ *"undefeated"* then all of the immediate ancestors of b are also assigned *"undefeated"*. By repeated application of this implication, all the ancestors of b are assigned *"undefeated"*, so $\sigma(c) =$ *"undefeated"*, which is a contradiction.

3. Let $a \in A$ be such that $+ \in l(a)$, and let $b \in A$ be such that $b \rightsquigarrow a$. Since $+ \in l(a)$, either $\sigma(a) =$ *"undefeated"* or $\sigma(a) = \emptyset$. Suppose first that $\sigma(a) =$ *"undefeated"*. Since $b \rightsquigarrow a$, either $b \rightharpoonup a$ or there is an ancestor c of a such that $b \rightharpoonup c$. In the former case, $\sigma(b) =$ *"defeated"*. In the latter case $\sigma(c) =$ *"defeated"* so $\sigma(b) =$ *"defeated"*. Thus, in both cases $l(b) = -$, so condition 3 of definition 3 is satisfied. Suppose now that $\sigma(a) = \emptyset$. This means that $l(a) = \pm$. Again, since $b \rightsquigarrow a$, either $b \rightharpoonup a$ or there is an ancestor c of b such that $c \rightharpoonup a$. In the former case, we cannot have $\sigma(b) =$ *"undefeated"*, since this would imply that $\sigma(a) =$ *"defeated"*, which is not the case. Thus, $l(b) \ni -$. Similarly, in the latter case we cannot have $\sigma(c) =$ *"undefeated"*. Therefore, we cannot have $\sigma(b) =$ *"undefeated"*, since this would imply that all of the immediate ancestors of b are also assigned *"undefeated"* and, by repeated application of this reasoning, that all ancestors of b are assigned *"undefeated"*. This would mean that $\sigma(c) =$ *"undefeated"*, which is a contradiction. Thus, $\sigma(b) \neq$ *"undefeated"*, so $l(b) \ni -$.

Thus, l is a labeling. Clearly, l is complete. □

Theorem 17: Let G, AF, σ and l be as defined in theorem 16. The mapping σ is a status assignment iff the mapping l corresponds to a preferred set of arguments (i.e. iff there is a preferred set S of arguments such that $S = l^+$).

Proof: Suppose that σ is a status assignment. By theorem 16 and definition 24, l is a rooted complete labeling such that there is no other rooted complete labeling which is a strict refinement of l. Notice that l^+ is maximal rooted since, otherwise, there is a rooted complete labeling l' of AF such that $l^+ \subset l'^+$. This would mean that $\{a \in A \mid l^+ \rightsquigarrow a\} \subseteq \{a \in A \mid l'^+ \rightsquigarrow a\}$. Since l and l' are rooted, this means that $l^- \subseteq l'^-$. Thus we would have $l^+ \subset l'^+$ and $l^- \subseteq l'^-$, which would mean that l' is a strict refinement of l. Thus, l^+ is indeed maximal rooted. By theorem 15, then, l^+ is preferred.

Now suppose that l is a labeling such that l^+ is preferred. By theorem 15 l^+ is maximal completely rooted. Notice that the complete labeling corresponding to a completely rooted set is uniquely defined. (This is not true for general sets of arguments.)

Therefore, l is a rooted complete labeling. In order to show that there is no other rooted complete labeling which is a strict refinement of l, let $l' \sqsubseteq l$ be a rooted complete labeling. Since $l' \sqsubseteq l$ we have $l'^{+} \supseteq l^{+}$. Notice that we cannot have $l'^{+} \supset l^{+}$ because this would contradict the maximality of l. Thus, $l'^{+} = l^{+}$. Therefore, $\{a \in A \mid l'^{+} \rightsquigarrow a\} = \{a \in A \mid l^{+} \rightsquigarrow a\}$, so $l'^{-} = l^{-}$. Thus, $l' = l$. By theorem 16 and definition 24, then, σ is a status assignment. □

Theorem 18: Let $AF = (A, \rightsquigarrow)$ be an argumentation framework. The set of obligatory arguments of AF is completely acceptable.

Proof: Let S be the set of obligatory arguments of AF. Define the mapping l as follows: let

$$\begin{aligned} l^{+} &= S \\ l^{-} &= \{a \in A \mid a \rightsquigarrow S \text{ or } S \rightsquigarrow a\} \\ l^{\pm} &= A \backslash l^{+} \cup l^{-} \end{aligned}$$

We show that the mapping l is a labeling by verifying the three conditions of definition 3 as follows:

1. Let $a \in A$ be such that $l(a) \ni -$. This means that $a \notin S$, so there is a maximal robust set S' such that $a \notin S'$. Let l' be a robust labeling that corresponds to S'. Since $a \notin S'$ we have $l'(a) \neq +$. Notice that there must be an argument $b \rightsquigarrow a$ such that $l'(b) \neq -$. Indeed, suppose on the contrary that $\forall b \rightsquigarrow a$, $l'(b) = -$. Since $l'(a) \neq +$ we must have $l'(a) = \emptyset$. In this case the mapping l'' defined as follows, would be a well-defined labeling:

   $$l''(c) = \begin{cases} + & \text{if } c = a \\ - & \text{if } a \rightsquigarrow c \\ l'(c) & \text{otherwise} \end{cases}$$

   In addition, it is not difficult to see that, since l' is robust, l'' is also robust. Notice that l'' corresponds to the robust set $S' \cup \{a\}$. This implies that S' is not a maximal robust set, which is a contradiction. Thus, we have shown that there is indeed an argument $b \rightsquigarrow a$ such that $l'(b) \neq -$. This implies that $b \not\rightsquigarrow l'^{+}$ and $l'^{+} \not\rightsquigarrow b$. Since $S \subseteq l'^{+}$ we then have $b \not\rightsquigarrow S$ and $S \not\rightsquigarrow b$. By definition of l, then, $l(b) \neq -$, so $l(b) \ni +$.

2. Let $a \in A$ be such that $l(a) \ni +$ and let $b \rightsquigarrow a$. To show that $l(b) \ni -$ suppose, on the contrary, that $l(b) = +$. This would mean that $b \in S$, which would imply that $S \rightsquigarrow a$, so $l(a) = -$, which is a contradiction.

3. Let $a, b \in A$ be such that $l(a) \ni +$ and $a \rightsquigarrow b$. To show that $l(b) \ni -$ suppose, on the contrary, that $l(b) = +$. This would mean that $b \in S$, which would imply that $a \rightsquigarrow S$, so $l(a) = -$, which is a contradiction.

Clearly, the labeling l is complete. □

References

[BDKT97] A. Bondarenko, P.M. Dung, R. A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1–2):63–101, 1997.

[BG94] G. Brewka and T. Gordon. How to buy a porsche: An approach to defeasible decision making. In *Working Notes of AAAI-94 Workshop on Computational Dialectics*, pages 28–38, Seattle, July 1994.

[Dun95] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

[GK96] T. Gordon and N. Karacapilidis. The zeno argumentation framework. In *Proceedings of the biannual International Conference on Formal and Applied Practical Reasoning (FAPR) workshop*, 1996. available at http://nathan.gmd.de/projects/zeno/fapr/programme.html.

[Jak95] H. Jakobovits. A theory of argumentation and its application to semantics for logic programs. Msc thesis, Free University of Brussels, VUB, 1995.

[JV96] H. Jakobovits and D. Vermeir. Contradiction in argumentation frameworks. In *Proceedings of the IPMU conference*, pages 821–826, 1996.

[JV97] H. Jakobovits and D. Vermeir. Argumentation and logic programming semantics. in preparation, December 1997.

[KMD94] A. C. Kakas, P. Mancarella, and Phan Minh Dung. The acceptability semantics for logic programs. In P. Van Hentenrijck, editor, *Proceedings of the 11th International Conference on Logic Programming*, pages 504–519. MIT Press, 1994.

[KPG96] N. Karacapilidis, D. Papadias, and T. Gordon. An argumentation based framework for defeasible and qualitative reasoning. In D.L. Borges and C.A.A. Kaestner, editors, *Advances in Artificial Intelligence, Proceedings of the XIIIth Brazilian Symposium on Artificial Intelligence*, volume 1159, pages 1–10, Curitiba, Brazil, 1996. Springer-Verlag.

[KT96] Robert A. Kowalski and Francesca Toni. Abstract argumentation. *Artificial Intelligence and Law Journal, Special Issue on Logical Models of Argumentation*, 4, 1996.

[Men79] E. Mendelson. *Introduction to Mathematical Logic*. Van Nostrand, second edition edition, 1979.

[Pol94] John Pollock. Justifi cation and defeat. *Artificial Intelligence*, 67:377–407, 1994.

[Pra96] H. Prakken. Dialectical proof theory for defeasible argumentation with defeasible priorities. In *Proceedings of the biannual International Conference on Formal and Applied Practical Reasoning (FAPR) workshop*, 1996. available at http://nathan.gmd.de/projects/zeno/fapr/programme.html.

[PS95] H. Prakken and G. Sartor. On the relation between legal language and legal argument: assumptions, applicability and dynamic priorities. In *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, pages 1–9. ACM Press, 1995.

[PS96] H. Prakken and G. Sartor. A system for defeasible argumentation, with defeasible priorities. In *Proceedings of the International Conference on Formal Aspects of Practical Reasoning*, Springer Lecture Notes in AI 1085, pages 510–524, Bonn, 1996. Springer Verlag.

[Sed90] R. Sedgewick. *Algorithms in C*. Addison-Wesley Publishing Company, 1990.

[SZ90] D. Sacca and C. Zaniolo. Stable models and non-determinism for logic programs with negation. In *Proceedings of the 9th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 205–218. Association for Computing Machinery, 1990.

[Tar55] A. Tarski. A lattice theoretical fi xpoint theorem and its application. *Pacific Journal of Mathematics*, 5:285–309, 1955.

[Vre97] Gerard A.W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90(1-2):225–279, 1997.