Checking the acceptability of a set of arguments

Philippe Besnard and **Sylvie Doutre**
IRIT – Université Paul Sabatier
118, route de Narbonne
31062 Toulouse Cedex 4 – France
{besnard, doutre}@irit.fr

Abstract

Considering Dung's argumentation framework and semantics, we are interested in the problem which consists in deciding whether a set of arguments is acceptable under a given semantics. We look at three approaches. The first one consists in testing whether the set satisfies an equation; In particular, we look at the equations presented in (Dung 1995; Besnard & Doutre 2004). The second approach consists in testing whether the set is a model of a propositional formula and the third one consists in testing the satisfiability of a propositional formula.

Introduction

Argumentation is a reasoning model which amounts to building and evaluating arguments, generally conflicting. Dung's argumentation framework ((Dung 1995)) constitutes an adequate formal framework to study this reasoning model. Its abstract structure makes it possible to unify many other approaches proposed for argumentation on the one hand (see (Prakken & Vreeswijk 2002) for a synthesis of these approaches) and formalisms modelling non-monotonic reasoning on the other hand (see (Bondarenko *et al.* 1997; Doutre 2002)).

From a set of arguments and a binary relation between arguments representing the notion of a conflict, arguments are evaluated in order to determine the most acceptable ones. Among the semantics given to acceptability, those of Dung define sets of arguments jointly acceptable called extensions.

Algorithms to compute extensions are presented in (Doutre & Mengin 2001). The credulous decision problem (does an argument belong to at least one extension?) and the skeptical one (does a given argument belong to any extension?) associated to each semantics are studied in various works: Algorithms to answer these problems are presented in (Cayrol, Doutre, & Mengin 2003), dialectical proof procedures are introduced in (Jakobovits & Vermeir 1999; Vreeswijk & Prakken 2000; Cayrol, Doutre, & Mengin 2003) and algorithms computing some of these proof procedures are established in (Cayrol, Doutre, & Mengin 2003).

In this article, we are interested in the problem which consists in deciding whether a set is an extension of a given semantics.

For his well-known stable semantics, Dung answers this problem by giving in (Dung 1995) a simple equation that a set satisfies if and only if it is a stable extension. Equations of this kind for two other semantics and new equations for the stable semantics are established in (Besnard & Doutre 2004); These equations aim at exhibiting the sameness of the various notions of an extension as introduced by Dung. We are going to briefly study if some of them could be efficient to check if a set is an extension under a given semantics.

Then, to answer the problem at hand, we will turn to another technique already considered in other formalisms used to represent knowledge: it consists in associating to the formalism a formula in propositional logic whose models correspond to the acceptable sets of the formalism. For example, in (Ben-Eliyahu & Dechter 1996) a formula of propositional logic is given whose models correspond to the extensions of a default theory. A similar correspondence is established for circumscription in (Gelfond, Przymusinska, & Przymusinski 1989) and for disjunctive logic programs in (Ben-Eliyahu & Dechter 1994). In the same way, we attempt to associate to an argument system a propositional formula whose models correspond to the extensions under a given semantics. For the stable semantics, the work of (Creignou 1995) completed in graph theory can be used. Notice that in (Dung 1995) a logic program is associated to an argument system and that the stable models of this logic program correspond to the stable extensions of the argument system; A translation of this logic program into propositional logic would associate to the argument system a propositional formula whose models would be the stable extensions of the system. This association is in two steps and could lead to formulas containing more informations than necessary; In this article we want to make such an association directly from a Dung argument system.

Exploiting again propositional logic, we will study a third way to check if a set is an extension. This way consists in associating to the set a propositional formula which is satisfiable if and only if the set is an extension of the considered semantics. This method and the previous one would make possible the use of existing constraint satisfaction and satisfiability techniques possible for argumentation.

This article presents preliminary ideas in each of the three approaches explicited above. The outline of the paper is as follows: In the next section we present Dung's argumentation framework and semantics. In the section "Equation checking", we study the equations characterizing extensions established in (Dung 1995; Besnard & Doutre 2004) to decide if a set is an extension of a given semantics. In the section "Model checking", we show how to associate to an argument system a propositional formula such that the models of the formula correspond to the extensions of the system under a given semantics. In the section "Satisfiability checking", we attempt to associate to the set which one wants to know if it is acceptable a propositional formula satisfiable if and only if the set is acceptable.

Argumentation and extensions

The argument system defined by Dung in (Dung 1995) is an abstract system in which arguments and conflicts between arguments are primitives.

Definition 1 *(Dung 1995) An* argument system *is a pair* (A, R) *where* A *is a set of* arguments *and* R *is a binary relation over* A *which represents a notion of* attack *between arguments (*$R \subseteq A \times A$*). Given two arguments* a *and* b*,* $(a, b) \in R$ *or equivalently* aRb*, means that* a attacks b *or that* a *is an* attacker *of* b*. A set of arguments* S *attacks an argument* a *if* a *is attacked by an argument of* S*. A set of arguments* S *attacks a set of arguments* S' *if there is an argument* $a \in S$ *which attacks an argument* $b \in S'$*.*

In all the definitions and notations which follow, we assume that an argument system (A, R) is given.

An argument system can be represented in a very simple way by a directed graph whose vertices are the arguments and edges correspond to the elements of R.

Dung gave several *semantics* to acceptability. These various semantics produce none, one or several acceptable sets of arguments, called *extensions*. One of these semantics, the stable semantics, is only defined via the notion of an attack:

Definition 2 *(Dung 1995) A set* $S \subseteq A$ *is* conflict-free *iff it does not exist two arguments* a *and* b *in* S *such that* a *attacks* b*. A conflict-free set* $S \subseteq A$ *is a* stable extension *iff for each argument which is not in* S*, there exists an argument in* S *that attacks it.*

The other semantics for acceptability rely upon the concept of defense:

Definition 3 *An argument* a *is* defended *by a set* $S \subseteq A$ *(or* S *defends* a*) iff for any argument* $b \in A$*, if* b *attacks* a *then* S *attacks* b*.*

An acceptable set of arguments according to Dung must be a conflict-free set which defends all its elements. Formally:

Definition 4 *(Dung 1995) A conflict-free set* $S \subseteq A$ *is* admissible *iff each argument in* S *is defended by* S*.*

Even if the definition of a stable extension does not rely upon the notion of defense, a stable extension is an admissible set. Admissibility has an advantage over stable semantics: given an argument system, there need not be any stable extension but there always exists at least one admissible set (the empty set is always admissible). A drawback of admissibility is that an argument system may have a large number of admissible sets. This is why other notions of acceptability which select only some admissible sets were introduced. Besides the stable semantics, three semantics refining admissibility have been introduced by Dung:

Definition 5 *(Dung 1995) A* preferred extension *is a maximal (wrt set inclusion) admissible subset of* A*. An admissible* $S \subseteq A$ *is a* complete extension *iff each argument which is defended by* S *is in* S*. The least (wrt set inclusion) complete extension is the* grounded extension*.*

Notice that a stable extension is also a preferred extension and a preferred extension is also a complete extension. Stable, preferred and complete semantics admit multiple extensions whereas the grounded semantics ascribes a single extension to a given argument system.

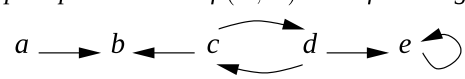
Deciding if a set is a stable extension or an admissible set can be computed in polynomial time, but deciding if a set is a preferred extension is a CO-NP-complete problem. These results of complexity were given in (Dimopoulos & Torres 1996) in the context of graph theory; given that the concepts of kernel, semi-kernel and maximum semi-kernel correspond respectively to the concepts of stable extension, admissible set and preferred extension (see (Dunne & Bench-Capon 2002; Doutre 2002)), these results are transposable to argumentation.

Example 1 *Let* (A, R) *be the argument system such that*

$$A = \{a, b, c, d, e\} \textit{ and}$$

$$R = \{(a, b), (c, b), (c, d), (d, c), (d, e), (e, e)\}.$$

The graph representation of (A, R) *is the following:*



The admissible sets of (A, R) *are* $\emptyset$, $\{a\}$, $\{c\}$, $\{d\}$, $\{a, c\}$ *and* $\{a, d\}$*. Dung's semantics induce the following acceptable sets:*

- *Stable extension(s):* $\{a, d\}$
- *Preferred extensions:* $\{a, c\}$, $\{a, d\}$
- *Complete extensions:* $\{a, c\}$, $\{a, d\}$, $\{a\}$
- *Grounded extension:* $\{a\}$.

Equation checking

A first method to check if a set is acceptable under a given semantics is to check if the set satisfies a given equation. Since Dung's semantics rely upon the notion of admissibility, it is natural to express these equations in terms of attack and defense.

An equation of this kind is presented by Dung himself for the stable extensions in (Dung 1995). Prior to giving this result, some notations are introduced: Given an argument system (A, R), for every set $S \subseteq A$:

$$\overline{S} \stackrel{\text{def}}{=} A \setminus S$$

$$Def(S) \stackrel{\text{def}}{=} \{a \in A \mid S \text{ defends } a\}$$

$$R^+(S) \stackrel{\text{def}}{=} \{a \in A \mid S \text{ attacks } a\}$$

$$R^-(S) \stackrel{\text{def}}{=} \{a \in A \mid a \text{ attacks an argument of } S\}$$

Proposition 1 *(Dung 1995) Given an argument system* (A,R)*,* $S \subseteq A$ *is a stable extension iff* $S = \overline{R^+(S)}$*.*

According to Proposition 1, checking if a set S is a stable extension amounts to computing the set of arguments which are not attacked by S and then testing if this set is equal to S.

The equation presented in Proposition 1 is clearly more efficient than the ones presented in (Besnard & Doutre 2004), where one has to compute the set of arguments which are not attacked by S and the set of argument which do not attack S, and then to take their intersection or conjunction in order to find an X and a Y appearing in the equation characterizing stable extensions:

Proposition 2 *(Besnard & Doutre 2004) Let* (A,R) *be an argument system. Let* $S \subseteq A$*. For all* $X \subseteq A$ *and for all* $Y \subseteq A$ *such that at least one of the conditions below is satisfied*

1. $\overline{R^+(S)} \cap \overline{R^-(S)} \subseteq X \subseteq \overline{R^-(S)}$ *and* $\overline{R^+(S)} \cap \overline{R^-(S)} \subseteq Y \subseteq \overline{R^+(S)} \cup \overline{R^-(S)}$
2. $\overline{R^+(S)} \cap \overline{R^-(S)} \subseteq X$ *and* $\overline{R^+(S)} \cap \overline{R^-(S)} \subseteq Y \subseteq \overline{R^-(S)}$

the following property holds:

$$S \text{ is a stable extension iff } S = Def((S \cup X) \cap Y).$$

Whereas Dung does not give any equation characterizing his other semantics, (Besnard & Doutre 2004) presents equations characterizing the preferred and the complete extensions. These equations are driven by the aim of exhibiting the sameness of Dung's extensions.

(Besnard & Doutre 2004)'s equations characterizing the preferred semantics are not effective since one has to know whether a set S is a preferred extension before giving one of these equations. An effective equation characterizing preferred extensions remains to be exhibited.

Concerning the complete extensions, (Besnard & Doutre 2004) presents a simple equation:

Proposition 3 *(Besnard & Doutre 2004) Given* (A,R)*,*

$$S \subseteq A \text{ is a complete extension iff } S = Def(S) \cap \overline{R^+(S)}.$$

According to this Proposition, checking if a set S is a complete extension amounts to computing the set of arguments defended by S, the set of arguments not attacked by S and then to test if their intersection is equal to S.

The equation of Proposition 3 is more efficient than the other equations also presented in (Besnard & Doutre 2004), where one has to first compute $Def(S) \cap \overline{R^+(S)}$ in order to find a set X appearing in the equation characterizing complete extensions:

Proposition 4 *(Besnard & Doutre 2004) Let* (A,R) *be an argument system. Let* $S \subseteq A$*. For all* $X \subseteq A$ *and for all* $Y \subseteq A$ *such that*

- $X \subseteq Def(S) \cap \overline{R^-(S)}$ *or* $X \subseteq Def(S) \cap \overline{R^+(S)}$ *and*
- $\overline{R^-(S)} \cap \overline{R^+(S)} \subseteq Y \subseteq \overline{R^-(S)} \cup \overline{R^+(S)}$

the following holds:

$$\begin{aligned} S \text{ is a complete extension} \quad &\text{iff} \quad S = Def(S \cup X) \cap Y \\ &\text{iff} \quad S = Def((S \cup X) \cap Y). \end{aligned}$$

(Besnard & Doutre 2004) suggests that these last equations could be used in order to speed up refuting that a set S is a complete extension. It is actually the case if one chooses Y to be the set of arguments not attacked by S (or the set of arguments which do not attack S) and X to be the set of arguments which have no attacker in (A,R); $Def(S) \cap \overline{R^+(S)}$ and $Def(S) \cap \overline{R^-(S)}$ do not have to be computed in order to check if such an X is included in them, it is always the case; If $X \not\subseteq S$, then X prevents the equations to hold.

Concerning the grounded extension, an equation remains to be found.

Model checking

A second technique to check if a set S is an extension of an argument system consists in characterizing all the extensions of the system by the models of a formula expressed in propositional logic; S is an extension if and only if S corresponds to a model of the formula. This method only works for finite sets of arguments.

Given an argument system (A,R), let us consider the propositional language $\mathcal{L}$ whose propositional symbols are the elements of A. For a given semantics, we will associate to (A,R) a formula whose models (maximum or minimal in certain cases), will correspond to the acceptable sets under the semantics. A model is represented by the set of the atoms which it satisfies, in other words a model of a formula is a set included in A.

Such a characterization already exists for the stable semantics. This characterization is not given in the context of argumentation but in the context of graph theory by (Creignou 1995). We already underlined in the section "Argumentation and extensions" the correspondence between the two contexts. Consequently, characterizing by a logical formula the kernels of a graph, (Creignou 1995) characterizes the stable extensions of an argument system.

Proposition 5 *Let* (A,R) *be an argument system. A set* $S \subseteq A$ *is a stable extension iff* S *is a model of the formula*

$$\bigwedge_{a \in A} (a \leftrightarrow \bigwedge_{b:(b,a) \in R} \neg b).$$

In order to characterize the other semantics, we are first going to associate to an argument system (A,R) a formula whose models are the admissible sets of (A,R). Let us recall that an admissible set is conflict-free and that it defends all its elements.

First, it is known that if an argument $a \in A$ is attacked by an argument $b \in A$, then a set containing a will be conflict-free only if it does not contain b. The formula

$$\Phi_1 = \bigwedge_{a \in A} (a \rightarrow \bigwedge_{b:(b,a) \in R} \neg b)$$

admits as its models the conflict-free sets of (A,R). This formula is not the only one. Indeed, one could also express the fact that if an argument $a \in A$ attacks an argument $b \in A$ then a set containing a is conflict-free only if it does not contain b. The formula

$$\Phi_1' = \bigwedge_{a \in A} (a \rightarrow \bigwedge_{b:(a,b) \in R} \neg b)$$

also admits as its models the conflict-free sets of (A, R). One can also view the concept of conflict-freedom like the fact that for any pair $(a, b) \in R$, a and b cannot both belong to a set if this set is to be conflict-free. Thus, the formula

$$\Phi_1'' = \bigwedge_{(a,b) \in R} (\neg a \vee \neg b)$$

also admits as its models the conflict-free sets of (A, R).

Second, in order to characterize the sets which defend all their elements, a formula has to capture the idea that if an argument $a \in A$ belongs to a set which defends all its elements, then for each of its attackers $b \in A$, there must be in the set an element c which attacks b. The formula

$$\Phi_2 = \bigwedge_{a \in A} (a \rightarrow \bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c))$$

admits as its models the sets of (A, R) which defend all their elements.

Finally, a formula characterizing the admissible sets of (A, R) is the conjunction of a formula characterizing the conflict-free sets of (A, R) (Φ_1, Φ_1' or Φ_1'') and of the formula Φ_2. If one chooses the formula Φ_1 to characterize the conflict-free sets, the admissible sets are characterized as follows:

Proposition 6 *Let (A, R) be an argument system. A set $S \subseteq A$ is admissible iff S is a model of the formula*

$$\bigwedge_{a \in A} ((a \rightarrow \bigwedge_{b:(b,a) \in R} \neg b) \wedge (a \rightarrow \bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c))).$$

The preferred extensions are maximal admissible sets. They are thus characterized as follows:

Proposition 7 *Let (A, R) be an argument system. A set $S \subseteq A$ is a preferred extension iff S is a maximal model of the formula*

$$\bigwedge_{a \in A} ((a \rightarrow \bigwedge_{b:(b,a) \in R} \neg b) \wedge (a \rightarrow \bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c))).$$

A formula characterizing the complete semantics must capture the idea that any argument defended by an extension must belong to the extension. In other words, if an argument $a \in A$ is such that each of its attackers b has an attacker c which belongs to the extension, then a must belong to the extension. The formula

$$\bigwedge_{a \in A} ((\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)) \rightarrow a))$$

admits as its models the sets which contain all the arguments that they defend. A characterization of the complete extensions is thus the following one:

Proposition 8 *Let (A, R) be an argument system. A set $S \subseteq A$ is a complete extension iff S is a model of the formula*

$$\bigwedge_{a \in A} ((a \rightarrow \bigwedge_{b:(b,a) \in R} \neg b) \wedge (a \leftrightarrow \bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c))).$$

The grounded extension being the least complete extension, it can be characterized as follows:

Proposition 9 *Let (A, R) be an argument system. A set $S \subseteq A$ is the grounded extension iff S is the minimal model of the formula*

$$\bigwedge_{a \in A} ((a \rightarrow \bigwedge_{b:(b,a) \in R} \neg b) \wedge (a \leftrightarrow \bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c))).$$

A drawback of this technique is that one has to build a formula which can turn out to be quite large (since it is associated to the system (A, R)). However, this formula is built once for all and then can be used from one test to another.

Another technique consists in associating a formula to the set to test and then checking the satisfiability of this formula. The advantage is that the formula to be built may be less large than with the preceding technique, but a drawback is that for each test which one wishes to make, it is necessary to build a new formula. We present this technique in the following section.

Satisfiability checking

A third technique to check if a set S is an extension of an argument system consists in associating to S a formula in propositional logic; S is an extension under a given semantics if and only if the formula is satisfiable. This method only works for finite sets of arguments.

Let us consider an argument system (A, R) and a set $S \subseteq A$. One wants to know if S is conflict-free. It should for that be checked that, for any argument $a \in S$, the attackers b of a do not belong to S. In other words, it is necessary that the formula

$$\Psi_1 = \bigwedge_{a \in S} (a \wedge (\bigwedge_{b:(b,a) \in R} \neg b))$$

be satisfiable. One can also test if the formula

$$\Psi_1' = \bigwedge_{a \in S} (a \wedge (\bigwedge_{b:(a,b) \in R} \neg b))$$

is satisfiable in order to determine if S is conflict-free: In this case, we make sure that any argument attacked by $a \in S$ is out of S.

If one wants to check if S is such that any argument which does not belong to S is attacked by at least one argument of S, it is necessary that the formula

$$\Psi_2 = \bigwedge_{a \notin S} (\neg a \wedge (\bigvee_{b:(b,a) \in R} b))$$

be satisfiable.

To check if a set S is a stable extension, one has to test if the conjunction of one of the formulas testing conflict-freedom (Ψ_1 or Ψ_1') and of the formula Ψ_2 is satisfiable. If the formula Ψ_1 is chosen to test conflict-freedom, then:

Proposition 10 *Let (A, R) be an argument system. A set $S \subseteq A$ is a stable extension iff the formula*

$$\bigwedge_{a \in S} (a \wedge (\bigwedge_{b:(b,a) \in R} \neg b)) \wedge \bigwedge_{a \notin S} (\neg a \wedge (\bigvee_{b:(b,a) \in R} b))$$

is satisfiable.

To check if a set S defends all its elements, it should be verified that for any argument $a \in S$, each attacker b of a is attacked by an argument $c \in S$. This amounts to checking if the formula

$$\bigwedge_{a \in S} (\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)) \wedge \bigwedge_{a \notin S} \neg a$$

is satisfiable.

Thus, to test if a set S is admissible, one has to check if the conjunction of one of the formulas testing conflict-freedom and of the preceding formula, is a satisfiable formula.

Proposition 11 *Let (A, R) be an argument system. A set $S \subseteq A$ is admissible iff the formula*

$$\bigwedge_{a \in S} (a \wedge (\bigwedge_{b:(b,a) \in R} (\neg b \wedge (\bigvee_{c:(c,b) \in R} c)))) \wedge \bigwedge_{a \notin S} \neg a$$

is satisfiable.

To check if a set S is such that all the arguments which it defends belong to it, it should be made sure that for any argument $a \in A$, if each attacker b of a has an attacker $c \in S$ then $a \in S$. Hence the formula

$$\bigwedge_{a \in S} a \wedge \bigwedge_{a \notin S} \neg a \wedge \bigwedge_{a \in A} ((\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)) \rightarrow a)$$

must be satisfiable.

This formula can be simplified as follows:

$$\bigwedge_{a \in S} a \wedge \bigwedge_{a \notin S} \neg a \wedge \bigwedge_{a \notin S} \neg(\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)).$$

Testing if the conjunction of this formula and of the formula testing if S is admissible (Proposition 11) is satisfiable, one can check if S is a complete extension.

Proposition 12 *Let (A, R) be an argument system. A set $S \subseteq A$ is a complete extension iff the formula*

$$\bigwedge_{a \in S} (a \wedge (\bigwedge_{b:(b,a) \in R} \neg b) \wedge (\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)))$$
$$\wedge \bigwedge_{a \notin S} (\neg a \wedge \neg(\bigwedge_{b:(b,a) \in R} (\bigvee_{c:(c,b) \in R} c)))$$

is satisfiable.

To test if a set $S \subseteq A$ is a preferred extension, a formula which is satisfiable only when the set S is maximal among all the admissible sets of (A, R) has to be characterized. Such a formula is being studied, just like the formula to test if S is the grounded extension of (A, R): It should, in this last case, be made sure that the formula is satisfiable only when S is, among all the complete extensions of (A, R), the least one.

Conclusion

Given a Dung argument system, we were interested in this article in the problem which consists in determining if a set of arguments is acceptable under a given Dung's semantics. We have been interested in three approaches.

The first one consists in using an equation expressed in terms of defense and attack that a set satisfies if and only if it is acceptable under the given semantics. Concerning the stable semantics, the equation presented in (Dung 1995) is more efficient than the ones provided in (Besnard & Doutre 2004). Some equations characterizing the complete extensions presented in (Besnard & Doutre 2004) can be used to test if a set is a complete extension and others can be used to speed up refuting that a set is a complete extension. Equations testing if a set of arguments is a preferred or the grounded extension remain to be established.

The second approach consists in associating to an argument system a formula in propositional logic whose models are the acceptable sets under the given semantics. A set is acceptable if it is a model of the formula. We provided such formulas for all Dung's semantics.

The third approach consists in associating to the set to test a formula in propositional logic which is satisfiable if and only if the set is acceptable under the given semantics. Such formulas have been established for the stable and the complete extensions. They remain to be established for the preferred extensions and the grounded extension.

These two last approaches are preliminary to the use of satisfiability and constraint satisfaction techniques for argumentation. In particular, these techniques may be used to improve existing algorithms computing acceptable sets or answering decision problems.

References

Ben-Eliyahu, R., and Dechter, R. 1994. Propositional Semantics for Disjunctive Logic Programs. *Annals of Mathematics and Artificial Intelligence* 12(1–2):53–87.

Ben-Eliyahu, R., and Dechter, R. 1996. Default reasoning using classical logic. *Artificial Intelligence* 84:113–150.

Besnard, P., and Doutre, S. 2004. Characterization of semantics for argument systems. In *Ninth International Conference on the Principles of Knowledge Representation and Reasoning (KR 2004)*. To appear.

Bondarenko, A.; Dung, P.; Kowalski, R.; and Toni, F. 1997. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence* 93:63–101.

Cayrol, C.; Doutre, S.; and Mengin, J. 2003. On Decision Problems related to the preferred semantics for argumentation frameworks. *Journal of Logic and Computation* 13(3):377–403.

Creignou, N. 1995. The class of problems that are linearly equivalent to Satisfiability or a uniform method for proving NP-completeness. *Theoretical Computer Science* 145:111–145.

Dimopoulos, Y., and Torres, A. 1996. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science* 170:209–244.

Doutre, S., and Mengin, J. 2001. Preferred Extensions of Argumentation Frameworks: Computation and Query Answering. In Goré, R.; Leitsch, A.; and Nipkow, T., eds., *IJCAR 2001*, volume 2083 of *LNAI*, 272–288. Springer-Verlag.

Doutre, S. 2002. Autour de la sémantique préférée des systèmes d'argumentation. PhD Thesis, Paul Sabatier University, Toulouse.

Dung, P. 1995. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77:321–357.

Dunne, P., and Bench-Capon, T. 2002. Coherence in Finite Argument Systems. *Artificial Intelligence* 141(1–2):187–203.

Gelfond, M.; Przymusinska, H.; and Przymusinski, T. 1989. On the Relationship Between Circumscription and Negation as Failure. *Artificial Intelligence* 38(1):75–94.

Jakobovits, H., and Vermeir, D. 1999. Dialectic Semantics for Argumentation Frameworks. In *Proc. ICAIL'99*, 53–62. ACM Press.

Prakken, H., and Vreeswijk, G. 2002. *Handbook of Philosophical Logic*. Dordrecht/Boston/London: Kluwer Academic Publishers. chapter Logics for Defeasible Argumentation, 219–318.

Vreeswijk, G., and Prakken, H. 2000. Credulous and Sceptical Argument Games for Preferred Semantics. In *Proc. JELIA'2000*, 239–253. LNAI 1919.