Towards Constraints Handling by Conflict Tolerance in Abstract Argumentation Frameworks

Ofer Arieli
School of Computer Science
The Academic College of Tel-Aviv, Israel

Abstract

In this paper we incorporate integrity constraints in Dung-style abstract argumentation frameworks. We show that even for constraints of a very simple form, standard conflict-free semantics for argumentation frameworks are not adequate as conflicts among arguments should sometimes be accepted and tolerated. For this, we use conflict-tolerant semantics and show how corresponding extensions may be represented in terms of propositional formulas.

Introduction and Motivation

Dung's argumentation framework (1995) is a graph-style representation of what may be viewed as a dispute. It is instantiated by a set of abstract objects, called arguments, and a binary relation on this set that intuitively represents attacks between arguments. These structures have been found useful for modeling a range of formalisms for non-monotonic reasoning, including default logic (Reiter 1980), logic programming under stable model semantics (Gelfond and Lifschitz 1988), three-valued stable model semantics (Wu, Caminada, and Gabbay 2009) and well-founded model semantics (van Gelder, Ross, and Schlipf 1991), Nute's defeasible logic (Governatori et al. 2004), and so on.

Despite of their general nature, experience shows that in some cases argumentation frameworks lack sufficient expressivity for accurately capturing their domain, and some extra apparatus is needed to gain a more comprehensive representation. This observation motivated several works, like those of Amgoud and Cayrol (2002) and Modgil (2009), in which meta-knowledge, such as preferences relations among the arguments, is provided for refining the process of selecting the arguments that can collectively be accepted from the argumentation framework at hand.

In this paper we formalize the additional knowledge that is linked to argumentation frameworks in terms of *integrity constraints*, that is, conditions that every accepted set of arguments must satisfy. We show that the satisfaction of such constraints (and even very simple ones) sometimes requires to abandon the conflict-freeness assumption behind standard argumentation semantics, so it might happen that accepted arguments attack each other. Such a case is considered next.

Example 1 The phenomena of interference on one hand and the photoelectric effect on the other hand may stand behind conflicting arguments about whether light is a particle or a wave. Any choice between such arguments would obviously be arbitrary, and the dismissal of one of them would unavoidably yield erroneous conclusions about the nature of light. For having a realistic theory it is therefore essential in this case to adopt an attitude that tolerates both conflicting arguments.

To be able to capture situations like the one described in the example above we incorporate the conflicting-tolerant semantics described in (Arieli 2012). This also allows us to represent, using propositional languages, different kinds of semantics for argumentation frameworks, augmented with integrity constraints, and compute these semantics by off-the-shelf SAT-solvers.

The rest of this paper is organized as follows: First, we review the main definitions pertaining to Dung's theory of argumentation, then we show how integrity constrains can be added to this theory, and how four-valued labeling in the context of conflict-tolerant semantics can be incorporated for handling constrained argumentation. This is followed by a section in which we show that what is obtained is representable by signed theories whose models describe the intended semantics of the constrained argumentation frameworks. In the last section we conclude and consider some future work.

Preliminaries

Let us first recall the basics of abstract argumentation frameworks.

Definition 2 A (finite) *argumentation framework* (Dung 1995) is a pair $\mathcal{AF} = \langle \mathit{Args}, \mathit{Att} \rangle$, where Args (the set of arguments) is a finite set, and Att (the attack relation) is a relation on $\mathit{Args} \times \mathit{Args}$.

When $(A, B) \in \mathit{Att}$ we say that A *attacks* B (or that B is *attacked by* A). The set of arguments that are attacked by A is denoted by A^+ and the set of arguments that attack A is denoted by A^-. Now, for a set $\mathcal{S} \subseteq \mathit{Args}$ we denote, respectively, by $\mathcal{S}^+ = \bigcup_{A \in \mathcal{S}} A^+$ and by $\mathcal{S}^- = \bigcup_{A \in \mathcal{S}} A^-$ the set of arguments that are attacked by some argument in $\mathcal{S}$ and the set of arguments that attack some argument in $\mathcal{S}$. Accordingly, the set of arguments that are *defended by* $\mathcal{S}$ is

$\mathrm{Def}(\mathcal{S}) = \{A \in \mathit{Args} \mid A^- \subseteq \mathcal{S}^+\}$, that is, each attacker of an argument in this set is counter-attacked by (an argument in) $\mathcal{S}$.

The primary principles for accepting arguments are now defined as follows:

Definition 3 Let $\mathcal{AF} = \langle \mathit{Args}, \mathit{Att} \rangle$ be an argumentation framework and let $\mathcal{S} \subseteq \mathit{Args}$ be a set of arguments.

- $\mathcal{S}$ is *conflict-free* (with respect to $\mathcal{AF}$) iff $\mathcal{S} \cap \mathcal{S}^+ = \emptyset$.
- $\mathcal{S}$ is an *admissible extension* (of $\mathcal{AF}$) iff it is conflict free and $\mathcal{S} \subseteq \mathrm{Def}(\mathcal{S})$.
- $\mathcal{S}$ is a *complete extension* (of $\mathcal{AF}$) iff it is conflict free and $\mathcal{S} = \mathrm{Def}(\mathcal{S})$.

Thus, conflict-freeness assures that no argument in the set is attacked by another argument in the set, admissibility guarantees, in addition, that the set is self defendant, and complete sets are admissible ones that defend exactly themselves.

Example 4 Consider the framework $\mathcal{AF}_1$ of Figure 1. This framework has five admissible extensions: $\emptyset$, $\{A\}$, $\{B\}$, $\{A, C\}$ and $\{B, D\}$, three of them are complete: $\emptyset$, $\{A, C\}$ and $\{B, D\}$.

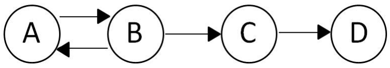

Figure 1: The argumentation framework $\mathcal{AF}_1$

Argument acceptability may now be defined as follows:

Definition 5 Let $\mathcal{AF} = \langle \mathit{Args}, \mathit{Att} \rangle$ be an argumentation framework. An argument $A \in \mathit{Args}$ is *credulously accepted* (by completeness semantics), if it belongs to *some* complete extension of $\mathcal{AF}$; it is *skeptically accepted* (by completeness semantics), if it belongs to *all* the complete extension of $\mathcal{AF}$.

Skeptical and credulous acceptance may be defined also with respect to other types of extensions, some of which are refinements of complete extensions. We refer, e.g., to (Baroni, Caminada, and Giacomin 2011) for further details.

Constrained Argumentation Frameworks

Consider again Example 1. Ignoring one of the (conflicting) phenomena described there means a partial and even misleading description of the situation at hand. It is therefore essential to accept *both* phenomena in any set of accepted arguments of the corresponding argumentation framework. In practice, this has two implications:

1. Some further, 'meta knowledge' about the arguments should be supplied and taken into account in the computation of extensions, and
2. contradictory conclusions should be maintained without reducing to triviality (that is, without accepting anything whatsoever in the presence of contradictions, as is the case, e.g., in classical logic).

The most straightforward way of supporting the second issue above is by lifting the conflict-freeness requirement in Definition 3, while keeping the other properties in the same definition. It follows that any argument in an extension must still be defended.

Definition 6 Let $\mathcal{AF} = \langle \mathit{Args}, \mathit{Att} \rangle$ be an argumentation framework and let $Ext \subseteq \mathit{Args}$.

- Ext is a *paraconsistently admissible* (or: *p-admissible*) extension for $\mathcal{AF}$, if $Ext \subseteq \mathrm{Def}(Ext)$.
- Ext is a *paraconsistently complete* (or: *p-complete*) extension for $\mathcal{AF}$, if $Ext = \mathrm{Def}(Ext)$.[1]

Thus, every admissible (respectively, complete) extension for $\mathcal{AF}$ is also p-admissible (respectively, p-complete) extension for $\mathcal{AF}$, but not the other way around.

Example 7 The argumentation framework $\mathcal{AF}_2$ that is shown in Figure 2 has two p-complete extensions: $\emptyset$ (which is also the only complete extension in this case), and $\{A, B, C\}$.

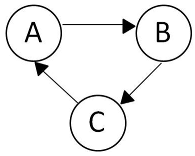

Figure 2: The argumentation framework $\mathcal{AF}_2$

As follows from Example 7, it may happen that the only (conflict-free) complete extension of a framework is the empty set. The next proposition shows that this is not the case as far as p-complete extensions are concerned.

Proposition 8 *There exists a nonempty p-complete extension (and so a nonempty p-admissible extension) for every argumentation framework.*[2]

Next, we extend the frameworks with integrity constraints that should be satisfied by any extension. In this paper we concentrate on constraints that can be expressed by single arguments. A natural requirement from such a set of constraints is that it should be p-admissible. This is so, since any accepted argument, not to mention those that *must* be accepted, has to be justified, and so such arguments shouldn't be exposed to undefended attacks. This leads to the next definition.

Definition 9 A *constrained argumentation framework* (CAF, for short) is a triple $\mathcal{CAF} = \langle \mathit{Args}, \mathit{Att}, \mathit{Const} \rangle$, where $\langle \mathit{Args}, \mathit{Att} \rangle$ is an argumentation framework, and Const (the set of *constraints*) is a p-admissible subset of Args.[3]

[1]The notions of p-admissibility and p-completeness should not be confused with similar notions, used in (Coste-Marquis, Devred, and Marquis 2005) for prudent semantics, which have a different meaning.

[2]Due to a lack of space proofs of some results are omitted.

[3]Alternatively, we shall sometimes refer to a constrained argumentation framework as a *pair* $\langle \mathcal{AF}, \mathit{Const} \rangle$, where $\mathcal{AF}$ is an argumentation framework and Const is a set of constraints.

Definition 10 An admissible (respectively, complete, p-admissible, p-complete) extension for a constrained argumentation framework $\mathcal{CAF} = \langle Args, Att, Const\rangle$ is a superset of $Const$, which is an admissible (respectively, complete, p-admissible, p-complete) extension of $\langle Args, Att\rangle$.

Example 11 Let $\mathcal{CAF}_1 = \langle Args, Att, Const\rangle$ be a constrained argumentation framework, where $\mathcal{AF}_1 = \langle Args, Att\rangle$ is the argumentation framework of Figure 1 and $Const = \{A, B\}$. This constrained framework does not have admissible nor complete extensions (since $Const$ is not conflict-free), but it has four p-admissible extensions: $\{A, B\}$, $\{A, B, C\}$, $\{A, B, D\}$ and $\{A, B, C, D\}$, the latter is also p-complete.

Proposition 12 *There exists a nonempty p-complete extension (and so a nonempty p-admissible extension) for every constrained argumentation framework.*

Proof. Let $\mathcal{CAF} = \langle Args, Att, Const\rangle$ be a constrained argumentation framework. If $Const = \emptyset$ then $\mathcal{CAF}$ is in fact an ('ordinary') argumentation framework, and so the proposition follows from Proposition 8. Suppose then that $Const \neq \emptyset$. By its definition, $Const$ is a p-admissible extension of $\mathcal{CAF}$. Now, if $Const$ is also a p-complete extension of $\mathcal{CAF}$, we are done. Otherwise, there is an argument $A_1 \in \mathrm{Def}(Const) - Const$. Let $Const_1 = Const \cup \{A_1\}$. Note that $Const_1$ is still p-admissible, since $A_1 \in \mathrm{Def}(Const)$, and so $Const_1 = Const \cup \{A_1\} \subseteq \mathrm{Def}(Const) \subseteq \mathrm{Def}(Const_1)$. Now, if $Const_1$ is p-complete, we are done. Otherwise, we again choose an argument $A_2 \in \mathrm{Def}(Const_1) - Const_1$, and consider the set $Const_2 = Const_1 \cup \{A_2\}$. As before, $Const_2$ is still p-admissible. By this process we get a sequence of p-admissible extensions $Const, Const_1, Const_2, \ldots$, where each extension is properly included in its successor. Note that this sequence consists of no more than $|Args|$ p-admissible sets, and it must culminate with a p-complete extension of $\mathcal{CAF}$. This is so, since if we keep adding arguments without reaching a p-complete extension, we eventually end-up with the whole set of arguments, $Args$, which must be a p-complete extension of $\mathcal{CAF}$.[4] □

Note 13 The p-complete extension of $\mathcal{CAF}$ constructed in the proof above is minimal in the sense that every set that is properly contained in it is not p-complete or does not contain the set $Const$. In this respect, we have shown that $\mathcal{CAF}$ has what may be called a "p-grounded extension".

Four-Valued Semantics for CAFs

For computing extensions of constrained argumentation frameworks we use the conflict-tolerant semantics for argumentation frameworks, introduced in (Arieli 2012). This approach is based on the following four-valued functions ('labelings') on the set of arguments: a value t assigned to an argument indicates that the argument should be accepted, f indicates that the argument should be rejected, $\top$ indicates that there are both supportive and rejective evidences, and $\bot$ is a no-acceptance no-rejection indication.

[4]Indeed, since the sequence contains only p-admissible extensions it holds, in particular, that $Args \subseteq \mathrm{Def}(Args)$. Also, obviously $\mathrm{Def}(Args) \subseteq Args$, thus $Args = \mathrm{Def}(Args)$.

Definition 14 Let $\mathcal{AF} = \langle Args, Att\rangle$ be an argumentation framework.

- Given a set $Ext \subseteq Args$ of arguments, the function that is *induced by* (or, is *associated with*) Ext is the 4-valued labeling $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ of $\mathcal{AF}$,[5] defined for every $A \in Args$ as follows:

$$p\mathcal{EL}_{\mathcal{AF}}(Ext)(A) = \begin{cases} t & \text{if } A \in Ext \text{ and } A \notin Ext^+ \\ \top & \text{if } A \in Ext \text{ and } A \in Ext^+ \\ f & \text{if } A \notin Ext \text{ and } A \in Ext^+ \\ \bot & \text{if } A \notin Ext \text{ and } A \notin Ext^+ \end{cases}$$

  A 4-valued labeling induced by some subset of $Args$ is called a *paraconsistent labeling* (or a *p-labeling*) of $\mathcal{AF}$.

- Given a 4-valued labeling lab of $\mathcal{AF}$, the set of arguments that is *induced by* (or, is *associated with*) lab is defined by

$$p\mathcal{LE}_{\mathcal{AF}}(lab) = \{A \mid lab(A) = t\} \cup \{A \mid lab(A) = \top\}.$$

The intuition behind the transformation from a labeling lab to its extension $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is that any argument for which there is some supportive indication (i.e., it is labeled t or $\top$) should be included in the extension (even if there are also opposing indications). The transformation from an extension Ext to the labeling $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ that it induces is motivated by the aspiration to accept the arguments in the extension by marking them as either t or $\top$. Since Ext is not necessarily conflict-free, two labels are required to indicate whether the argument at hand is attacked by another argument in the extension, or not.

In order to obtain p-admissible extensions of a CAF from its p-labelings, we pose further rationality postulates on p-labelings:

Definition 15 Let $\mathcal{AF} = \langle Args, Att\rangle$ be an argumentation framework. A p-labeling lab for $\mathcal{AF}$ is called *p-admissible*, if it satisfies the following rules:

pIn	If $lab(A) = t$ then $\forall B \in A^-\ lab(B) = f$.
pOut	If $lab(A) = f$ then $\exists B \in A^-$ s.t. $lab(B) \in \{t, \top\}$
pBoth	If $lab(A) = \top$ then $\forall B \in A^-\ lab(B) \in \{f, \top\}$ and $\exists B \in A^-$ s.t. $lab(B) = \top$.
pNone	If $lab(A) = \bot$ then $\forall B \in A^-\ lab(B) \in \{f, \bot\}$.

Proposition 16 (Arieli 2012)

- *If Ext is a p-admissible extension of $\mathcal{AF}$ then the function $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a p-admissible labeling of $\mathcal{AF}$.*
- *If lab is a p-admissible labeling of $\mathcal{AF}$ then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is a p-admissible extension of $\mathcal{AF}$.*
- *The functions $p\mathcal{EL}_{\mathcal{AF}}$ and $p\mathcal{LE}_{\mathcal{AF}}$, restricted to the p-admissible labelings and the p-admissible extensions of $\mathcal{AF}$, are bijective, and are each other's inverse.*

[5]Here, $p\mathcal{EL}$ stands for a **p**araconsistent-based conversion of **e**xtensions to **l**abelings.

A similar one-to-one correspondence holds between p-complete extensions and p-labelings that satisfy the following postulates:

Definition 17 Let $\mathcal{AF} = \langle Args, Att \rangle$ be an argumentation framework. A p-labeling lab for $\mathcal{AF}$ is called *p-complete*, if it satisfies the following rules:

pIn$^+$ $lab(A) = t$ iff $\forall B \in A^-\ lab(B) = f$.

pOut$^+$ $lab(A) = f$ iff $\exists B \in A^-$ s.t. $lab(B) \in \{t, \top\}$ and $\exists B \in A^-$ s.t. $lab(B) \in \{t, \bot\}$.

pBoth$^+$ $lab(A) = \top$ iff $\forall B \in A^-\ lab(B) \in \{f, \top\}$ and $\exists B \in A^-$ s.t. $lab(B) = \top$.

pNone$^+$ $lab(A) = \bot$ iff $\forall B \in A^-\ lab(B) \in \{f, \bot\}$ and $\exists B \in A^-$ s.t. $lab(B) = \bot$.

Proposition 18 (Arieli 2012)

- *If Ext is a p-complete extension of $\mathcal{AF}$ then the function $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a p-complete labeling of $\mathcal{AF}$.*
- *If lab is a p-complete labeling of $\mathcal{AF}$ then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is a p-complete extension of $\mathcal{AF}$.*
- *The functions $p\mathcal{EL}_{\mathcal{AF}}$ and $p\mathcal{LE}_{\mathcal{AF}}$, restricted to the p-complete labelings and the p-complete extensions of $\mathcal{AF}$, are bijective, and are each other's inverse.*

Example 19 The four p-complete extensions of $\mathcal{CAF}_1$ in Example 11 and the corresponding p-complete labelings are represented in the table below.

lab	A	B	C	D	induced extension
1	t	f	t	f	$\{A, C\}$
2	f	t	f	t	$\{B, D\}$
3	$\bot$	$\bot$	$\bot$	$\bot$	$\{\}$
4	$\top$	$\top$	$\top$	$\top$	$\{A, B, C, D\}$

Representation of p-Complete Extensions

In this section we show that p-complete extensions can be represented (and computed) by logic-based theories. The idea is the following: by Proposition 18, for computing the p-extensions of a given CAF it is sufficient to formalize the postulates in Definition 17 for that CAF, and then compute the four-valued models of the theory that is obtained. Now, instead of conducting the computations in the context of four-valued semantics, a simple syntactic transformation is applied on the underlying theory for modeling it in the context of two-valued propositional semantics. What is obtained is called *signed theories*, and their two-valued models are associated with the p-complete extensions of the CAF. Below, we show how this is done.[6]

Let us first represent the postulates of Definition 17 more formally:

Definition 20 The set $\mathsf{pCMP}_{\mathcal{AF}}(x)$ of expressions for an argumentation framework $\mathcal{AF} = \langle Args, Att \rangle$ is shown in Figure 3.

[6] A similar process may be applied also for representing p-admissible extensions, using Proposition 16 and Definition 15.

In the expressions of $\mathsf{pCMP}_{\mathcal{AF}}(x)$, x is a variable (to be sequentially substituted by the elements of $Args$), and $\mathsf{att}(y, x)$ is replaced by the propositional constant t if $(y, x) \in Att$ (that is, if y attacks x in $\mathcal{AF}$), and otherwise $\mathsf{att}(y, x)$ is replaced by the propositional constant f. The intuitive meaning of the expressions $\mathsf{val}(x, v)$ is that the argument x is assigned the value v, for $v \in \{t, f, \bot, \top\}$ (see also Definition 23). It follows that we are in the context of 4-valued semantics, considered in the previous section, where t and f represent the classical truth values, and the other two values, denoted $\bot$ and $\top$, intuitively represent lack of information and contradictory information (respectively).

For representing constraints, we add to $\mathsf{pCMP}_{\mathcal{AF}}(x)$ the following set of formulas, assuring that any argument in the set of constraints is always accepted:

$$\mathsf{Const}(Args) = \{\mathsf{val}(x, t) \vee \mathsf{val}(x, \top) \mid x \in Const\}.$$

Altogether, we get the following theory:

Definition 21 Let $\mathcal{CAF} = \langle \mathcal{AF}, Const \rangle$ be a constrained argumentation framework for the argumentation framework $\mathcal{AF} = \langle Args, Att \rangle$, and let $\mathsf{pCMP}_{\mathcal{AF}}(x)$ be the set of expressions that is obtained from $\mathcal{AF}$ according to Figure 3. We denote by $\mathsf{pCMP}_{\mathcal{AF}}[A_i/x]$ the substitution of x in these expressions by an argument $A_i \in Args$. Now,

- $\mathsf{pCMP}(\mathcal{AF}) = \bigcup_{A_i \in Args} \mathsf{pCMP}_{\mathcal{AF}}[A_i/x]$.
- $\mathsf{pCMP}(\mathcal{CAF}) = \mathsf{pCMP}(\mathcal{AF}) \bigcup \mathsf{Const}(Args)$.

Example 22 Let us explicate the second expression in Figure 3 for the argumentation framework $\mathcal{AF}_1$ of Figure 1, where $x = A$. Since the only attacker of A in $\mathcal{AF}_1$ is B, we have that $\mathsf{att}(y, A) = \mathsf{t}$ when $y = B$ and $\mathsf{att}(y, A) = \mathsf{f}$ for any $y \neq B$. Thus, we have:
$\mathsf{val}(A, f) \supset (\mathsf{val}(B, t) \vee \mathsf{val}(B, \top)) \wedge (\mathsf{val}(B, t) \vee \mathsf{val}(B, \bot))$.
As we shall see shortly, the expressions $\mathsf{val}(x, v)$ are abbreviations of formulas that hold iff the truth value of x is v (and the connectives in the expression are interpreted as usual). Thus, the expression above may be replaced by the simpler expression $\mathsf{val}(A, f) \supset \mathsf{val}(B, t)$. By similar translations and rewriting considerations, the set $\mathsf{pCMP}(\mathcal{AF}_1)$ is equivalent to the following set:

$$\left\{ \begin{array}{ll} \mathsf{val}(A, t) \supset \mathsf{val}(B, f), & \mathsf{val}(A, f) \supset \mathsf{val}(B, t), \\ \mathsf{val}(B, t) \supset \mathsf{val}(A, f), & \mathsf{val}(B, f) \supset \mathsf{val}(A, t), \\ \mathsf{val}(C, t) \supset \mathsf{val}(B, f), & \mathsf{val}(C, f) \supset \mathsf{val}(B, t), \\ \mathsf{val}(D, t) \supset \mathsf{val}(C, f), & \mathsf{val}(D, f) \supset \mathsf{val}(C, t), \\ \mathsf{val}(A, \top) \supset \mathsf{val}(B, \top), & \mathsf{val}(A, \bot) \supset \mathsf{val}(B, \bot)), \\ \mathsf{val}(B, \top) \supset \mathsf{val}(A, \top), & \mathsf{val}(B, \bot) \supset \mathsf{val}(A, \bot)), \\ \mathsf{val}(C, \top) \supset \mathsf{val}(B, \top), & \mathsf{val}(C, \bot) \supset \mathsf{val}(B, \bot)), \\ \mathsf{val}(D, \top) \supset \mathsf{val}(C, \top), & \mathsf{val}(D, \bot) \supset \mathsf{val}(C, \bot)) \end{array} \right\}$$

Thus, for the constrained argumentation framework $\mathcal{CAF}_1$ of Example 11, we have that $\mathsf{pCMP}(\mathcal{CAF}_1)$ consists of the expressions above, together with the following two extra conditions:

$$\mathsf{val}(A, t) \vee \mathsf{val}(A, \top), \quad \mathsf{val}(B, t) \vee \mathsf{val}(B, \top).$$

$$\mathsf{val}(x,t) \supset \bigwedge_{y\in Args}\Big(\mathsf{att}(y,x) \supset \mathsf{val}(y,f)\Big),$$
$$\mathsf{val}(x,f) \supset \Big(\bigvee_{y\in Args}\Big(\mathsf{att}(y,x) \wedge \Big(\mathsf{val}(y,t) \vee \mathsf{val}(y,\top)\Big)\Big) \wedge \bigvee_{y\in Args}\Big(\mathsf{att}(y,x) \wedge \Big(\mathsf{val}(y,t) \vee \mathsf{val}(y,\bot)\Big)\Big)\Big),$$
$$\mathsf{val}(x,\top) \supset \Big(\bigwedge_{y\in Args}\Big(\mathsf{att}(y,x) \supset \Big(\mathsf{val}(y,f) \vee \mathsf{val}(y,\top)\Big)\Big) \wedge \bigvee_{y\in Args}\Big(\mathsf{att}(y,x) \wedge \mathsf{val}(y,\top)\Big)\Big),$$
$$\mathsf{val}(x,\bot) \supset \Big(\bigwedge_{y\in Args}\Big(\mathsf{att}(y,x) \supset \Big(\mathsf{val}(y,f) \vee \mathsf{val}(y,\bot)\Big)\Big) \wedge \bigvee_{y\in Args}\Big(\mathsf{att}(y,x) \wedge \mathsf{val}(y,\bot)\Big)\Big)$$

Figure 3: The expressions $\mathsf{pCMP}_{\mathcal{AF}}(x)$

As argument labeling may use four values, the *intended* semantics of the $\mathsf{val}(x,v)$ expressions mentioned previously is a four-valued one. This is also evident by the four-valued semantics that was associated with CAF in the previous section. In this respect, Belnap's well-known four-valued framework for computerized reasoning (Belnap 1977) naturally fits to our setting. It is defined by the distributive lattice $\mathcal{FOUR} = (\{t, f, \top, \bot\}, \leq)$ in which t and f are the maximal and the minimal elements (respectively), and $\top, \bot$ are intermediate elements that are $\leq$-incomparable. This structure has an order reversing involution $\neg$, for which $\neg t = f$, $\neg f = t$, $\neg\top = \top$ and $\neg\bot = \bot$. We shall denote the meet and the join of this lattice by $\wedge$ and $\vee$, respectively. The implication connective is defined as follows: $a \supset b = t$ if $a \in \{f, \bot\}$, and $a \supset b = b$ otherwise.[7] As in (Belnap 1977), we take the values t and $\top$ to be our 'designated elements', i.e., those that designate acceptable assertions.

Let now $\mathcal{L}$ be a propositional language consisting of a set of atomic formulas, $\mathsf{Atoms}(\mathcal{L})$. For switching to two-valued semantics (and then being able to use standard SAT-solver and other theorem provers for classical logic), we follow the approach in (Arieli 2007) and consider a *signed alphabet* $\mathsf{Atoms}^{\pm}(\mathcal{L})$ that consists of two symbols $p^{\oplus}, p^{\ominus}$ for each atom $p \in \mathsf{Atoms}(\mathcal{L})$. The language over $\mathsf{Atoms}^{\pm}(\mathcal{L})$ with the same connectives as those of $\mathcal{L}$ is denoted by $\mathcal{L}^{\pm}$.

Now we are ready to define the (two-valued) signed theory for computing the p-complete extensions of CAFs.

Definition 23 Denote by $\mathsf{pCMP}^{\pm}(\mathcal{AF}))$ the signed theories obtained from $\mathsf{pCMP}(\mathcal{AF}))$ by the following substitutions:

$\mathsf{val}(p,t) = p^{\oplus} \wedge \neg p^{\ominus}$,
$\mathsf{val}(p,f) = \neg p^{\oplus} \wedge p^{\ominus}$,
$\mathsf{val}(p,\top) = p^{\oplus} \wedge p^{\ominus}$,
$\mathsf{val}(p,\bot) = \neg p^{\oplus} \wedge \neg p^{\ominus}$.

The theory $\mathsf{pCMP}^{\pm}(\mathcal{CAF})$ is obtained from $\mathsf{pCMP}(\mathcal{CAF})$ in the same way.

Example 24 By Example 22, we have that $\mathsf{pCMP}^{\pm}(\mathcal{AF}_1)$, where $\mathcal{AF}_1$ is the argumentation framework of Figure 1, is the following:

$$\left\{\begin{array}{ll}
A^{\oplus} \wedge \neg A^{\ominus} \supset \neg B^{\oplus} \wedge B^{\ominus}, & \neg A^{\oplus} \wedge A^{\ominus} \supset B^{\oplus} \wedge \neg B^{\ominus}, \\
B^{\oplus} \wedge \neg B^{\ominus} \supset \neg A^{\oplus} \wedge A^{\ominus}, & \neg B^{\oplus} \wedge B^{\ominus} \supset A^{\oplus} \wedge \neg A^{\ominus}, \\
C^{\oplus} \wedge \neg C^{\ominus} \supset \neg B^{\oplus} \wedge B^{\ominus}, & \neg C^{\oplus} \wedge C^{\ominus} \supset B^{\oplus} \wedge \neg B^{\ominus}, \\
D^{\oplus} \wedge \neg D^{\ominus} \supset \neg C^{\oplus} \wedge C^{\ominus}, & \neg D^{\oplus} \wedge D^{\ominus} \supset C^{\oplus} \wedge \neg C^{\ominus}, \\
A^{\oplus} \wedge A^{\ominus} \supset B^{\oplus} \wedge B^{\ominus}, & \neg A^{\oplus} \wedge \neg A^{\ominus} \supset \neg B^{\oplus} \wedge \neg B^{\ominus}, \\
B^{\oplus} \wedge B^{\ominus} \supset A^{\oplus} \wedge A^{\ominus}, & \neg B^{\oplus} \wedge \neg B^{\ominus} \supset \neg A^{\oplus} \wedge \neg A^{\ominus}, \\
C^{\oplus} \wedge C^{\ominus} \supset B^{\oplus} \wedge B^{\ominus}, & \neg C^{\oplus} \wedge \neg C^{\ominus} \supset \neg B^{\oplus} \wedge \neg B^{\ominus}, \\
D^{\oplus} \wedge D^{\ominus} \supset C^{\oplus} \wedge C^{\ominus}, & \neg D^{\oplus} \wedge \neg D^{\ominus} \supset \neg C^{\oplus} \wedge \neg C^{\ominus}
\end{array}\right\}$$

Accordingly, $\mathsf{pCMP}^{\pm}(\mathcal{CAF}_1)$, where $\mathcal{CAF}_1$ is considered in Example 11, is $\mathsf{pCMP}^{\pm}(\mathcal{AF}_1)$ and the two constraints:

$$(A^{\oplus} \wedge \neg A^{\ominus}) \vee (A^{\oplus} \wedge A^{\ominus}),\ \ (B^{\oplus} \wedge \neg B^{\ominus}) \vee (B^{\oplus} \wedge B^{\ominus})$$

which are equivalent to $A^{\oplus}$ and $B^{\oplus}$ (respectively).

The next result shows that the signed theories considered previously indeed allow to compute the p-complete extensions of (constrained) argumentation frameworks.[8]

Theorem 25 *Let $\mathcal{L}$ be a propositional language whose atomic formulas are associated with the arguments of a (constrained) argumentation framework $\mathcal{AF}$ ($\mathcal{CAF}$), and let $\mathcal{L}^{\pm}$ be the corresponding signed language. For a two-valued valuation ν on* $\mathsf{Atoms}^{\pm}(\mathcal{L})$*, we denote:* $\mathsf{Accept}(\nu) = \{A \mid \nu(\mathsf{val}(A,t)) = 1$ or $\nu(\mathsf{val}(\mathrm{A},\top)) = 1\}$.[9] *Then:*

- *The set of the p-complete extensions of an argumentation framework $\mathcal{AF}$ is the same as the following set:* $\{\mathsf{Accept}(\nu) \mid \nu$ *is a model of* $\mathsf{pCMP}^{\pm}(\mathcal{AF})\}$.
- *The set of the p-complete extensions of a constrained argumentation framework $\mathcal{CAF}$ is the same as the set:* $\{\mathsf{Accept}(\nu) \mid \nu$ *is a model of* $\mathsf{pCMP}^{\pm}(\mathcal{CAF})\}$.

Outline of proof. First, the correctness of the transformation between four- and two-valued semantics is proved in (Arieli 2007). The function val in Definition 23 is defined there for arbitrary formulas in $\mathcal{L}$, and it is shown that for every two-valued valuation ν^2 on $\mathsf{Atoms}^{\pm}(\mathcal{L})$ there is a unique four-valued valuation ν^4 on $\mathsf{Atoms}(\mathcal{L})$, and for every four-valued valuation ν^4 on $\mathsf{Atoms}(\mathcal{L})$ there is a unique two-valued valuation ν^2 on $\mathsf{Atoms}^{\pm}(\mathcal{L})$, such that for every formula ψ in $\mathcal{L}$, $\nu^4(\psi) = x$ iff $\nu^2(\mathsf{val}(\psi,x)) = 1$. Now, it remains to

[7] As it is shown e.g. in (Arieli and Avron 1998), this connective indeed acts as an 'implication' in this context.

[8] Below we freely exchange an argument $A_i \in Args$, the propositional variable that represents A_i (with the same notation), and the corresponding signed variables $A_i^{\oplus}$, $A_i^{\ominus}$.

[9] Note that, in fact, $\mathsf{Accept}(\nu) = \{A \mid \nu(A^{\oplus}) = 1\}$.

show that the signed theories defined above faithfully represent p-complete extensions. This follows from the following facts:

- For every p-complete extension Ext of $\mathcal{AF}$ there is a model ν of $\mathsf{pCMP}^{\pm}(\mathcal{AF})$, such that $Ext = \mathsf{Accept}(\nu) = \{A \mid \nu(A^{\oplus}) = 1\}$ (and also $Ext^{+} = \{A \mid \nu(A^{\ominus}) = 1\}$).
- For every model ν of $\mathsf{pCMP}^{\pm}(\mathcal{AF})$ there is a p-complete extension Ext of $\mathcal{AF}$ such that $Ext = \mathsf{Accept}(\nu)$.

Similar facts hold for p-complete extensions of $\mathcal{CAF}$ and the models of $\mathsf{pCMP}^{\pm}(\mathcal{CAF})$. $\square$

Example 26 Consider again the theory $\mathsf{pCMP}^{\pm}(\mathcal{AF}_1)$ in Example 24. The two-valued models of this theory are given in the table below:

	$A^{\oplus}$	$A^{\ominus}$	$B^{\oplus}$	$B^{\ominus}$	$C^{\oplus}$	$C^{\ominus}$	$D^{\oplus}$	$D^{\ominus}$
ν_1	1	0	0	1	1	0	0	1
ν_2	0	1	1	0	0	1	1	0
ν_3	0	0	0	0	0	0	0	0
ν_4	1	1	1	1	1	1	1	1

Thus, $\mathsf{Accept}(\nu_1) = \{A, C\}$, $\mathsf{Accept}(\nu_2) = \{B, D\}$, $\mathsf{Accept}(\nu_3) = \{\}$ and $\mathsf{Accept}(\nu_4) = \{A, B, C, D\}$. These are exactly the p-complete extensions of $\mathcal{AF}_1$, as indeed guaranteed by Theorem 25. Note that only ν_4 is also a model of $\mathsf{pCMP}^{\pm}(\mathcal{CAF}_1)$, and indeed $\{A, B, C, D\}$ is the only p-complete extension of $\mathcal{CAF}_1$ (see Example 11).

Conclusion and Perspectives

In this paper we have considered situations in which additional knowledge in the form of integrity constraints may have to be taken into account in the computations of extensions for Dung's-style argumentation frameworks. The incorporation of constraints imply that contradictory arguments may have to be accepted, and so the conflict-freeness assumption, which is a keystone of the existing argumentation semantics, should be abandoned. It is shown that this can be achieved by conflict-tolerant semantics, and that extensions of constrained argumentation frameworks can be represented in terms of signed theories.

The addition of constraints to argumentation frameworks has also been considered by Coste-Marquis, Devred, and Marquis (2006).[10] In contrast to the present approach, Coste-Marquis, Devred, and Marquis require conflict freeness, and so neither of the constraints nor the extensions of the framework may be contradictory. This requirement implies that nonempty extensions may not be available for a constrained argumentation framework. Recall that in our case this cannot happen, as indicated in Proposition 12.

Our framework may be extended and improved in several ways. Future research involves the implementation of computerized tools for automatically computing extensions of constrained frameworks, and the incorporation of more complex constraints in those frameworks. It should be noted that by using methods like that in (Arieli 2007) for representing propositional formulas by signed formulas, the latter should not be too complicated, at least as long as constraints remain at the propositional level.[11]

[10] An anonymous reviewer is thanked for pointing this out.

[11] For instance, demanding that the acceptance of an argument A implies the acceptance of an argument B may be formalized by the introduction of the constraint $A \supset B$, which can be enforced by adding the signed formula $\neg A^{\oplus} \vee B^{\oplus}$ to the corresponding signed theory.

References

Amgoud, L., and Cayrol, C. 2002. Inferring from inconsistency in preference-based argumentation frameworks. *Journal of Automated Reasoning* 29(2):125–169.

Arieli, O., and Avron, A. 1998. The value of the four values. *Artificial Intelligence* 102(1):97–141.

Arieli, O. 2007. Paraconsistent reasoning and preferential entailments by signed quantified Boolean formulas. *ACM Transactions on Computational Logic* 8(3). Article 18.

Arieli, O. 2012. Conflict-tolerant semantics for argumentation frameworks. In *Proc. JELIA'12*, Lecture Notes in Computer Science 7519, 28–40. Springer.

Baroni, P.; Caminada, M.; and Giacomin, M. 2011. An introduction to argumentation semantics. *The Knowledge Engineering Review* 26(4):365–410.

Belnap, N. D. 1977. A useful four-valued logic. In Dunn, J. M., and Epstein, G., eds., *Modern Uses of Multiple-Valued Logics*. Reidel Publishing Company. 7–37.

Coste-Marquis, S.; Devred, C.; and Marquis, P. 2005. Prudent semantics for argumentation frameworks. In *Proc. ICTAI'05*, 568–572. IEEE Computer Society.

Coste-Marquis, S.; Devred, C.; and Marquis, P. 2006. Constrained argumentation frameworks. In *Proc. KR'06*, 112–122. AAAI Press.

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

Gelfond, M., and Lifschitz, V. 1988. The stable model semantics for logic programming. In *Proc. ICLP'88*, 1070–1080. MIT Press.

Governatori, G.; Maher, M. J.; Antoniou, G.; and Billington, D. 2004. Argumentation semantics for defeasible logic. *Journal of Logic and Computation* 14(5):675–702.

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

Reiter, R. 1980. A logic for default reasoning. *Artificial Intelligence* 13:81–132.

van Gelder, A.; Ross, K. A.; and Schlipf, J. S. 1991. The well-founded semantics for general logic programs. *Journal of the ACM* 38(3):620–650.

Wu, Y.; Caminada, M.; and Gabbay, D. M. 2009. Complete extensions in argumentation coincide with 3-valued stable models in logic programming. *Studia Logica* 93(1–2):383–403. Special issue: new ideas in argumentation theory.