Conflict-Tolerant Semantics for Argumentation Frameworks

Ofer Arieli

School of Computer Science, The Academic College of Tel-Aviv, Israel.
oarieli@mta.ac.il

Abstract. We introduce new kinds of semantics for abstract argumentation frameworks, in which, while all the accepted arguments are justified (in the sense that each one of them must be defended), they may still attack each other. The rationality behind such semantics is that in reality there are situations in which contradictory arguments coexist in the same theory, yet the collective set of accepted arguments is not trivialized, in the sense that other arguments may still be rejected.
To provide conflict-tolerant semantics for argumentation frameworks we extend the two standard approaches for defining coherent (conflict-free) semantics for argumentation frameworks: the extension-based approach and the labeling-based approach. We show that the one-to-one relationship between extensions and labelings of conflict-free semantics is carried on to a similar correspondence between the extended approaches for providing conflict-tolerant semantics. Thus, in our setting as well, these are essentially two points of views for the same thing.

1 Introduction and Motivation

An abstract argumentation framework consists of a set of (abstract) arguments and a binary relation that intuitively represents attacks between arguments. A semantics for such a structure is an indication which arguments can be collectively accepted. A starting point of all the existing semantics for abstract argumentation frameworks is that their set(s) of acceptable arguments must be conflict-free, that is: an accepted argument should not attack another accepted argument. This means, in particular, a dismissal of any self-referring argument and a rejection of any contradictory fragment of the chosen arguments. However, in everyday life it is not always the case that theories are completely coherent although each of their arguments provides a solid assertion, and so contradictory sets of arguments should sometimes be accepted and tolerated. Moreover, a removal of contradictory indications in such theories may imply a loss of information and may lead to erroneous conclusions.

In this paper, we consider a more liberal approach for argumentation semantics, adhering conflicting indications (and so inconsistencies). For this, we extend the two most standard approaches for defining semantics to abstract argumentation frameworks as follows:

- *Extension-Based Semantics.* Existing semantics that are defined by this method share two primary principles: admissibility and conflict-freeness (see, e.g., [2, 3]). The former principle, guaranteeing that an extension *Ext* 'defends' all of its elements (i.e., *Ext* 'counterattacks' each argument that attacks some $e \in Ext$), is preserved also in our framework, since otherwise acceptance of arguments would be an arbitrary choice. However, the other principle is lifted, since – as indicated above – we would like to permit, in some cases, conflicting arguments.
- *Labeling-Based Semantics.* We extend the traditional three-state labelings of arguments (accepted, rejected, undecided – see [5, 7]) by a fourth state, so now apart of accepting or rejecting an argument, we have *two* additional states, representing two opposite reasons for avoiding a definite opinion about the argument as hand: One ('none'), indicating that there is too little evidence for reaching a precise conclusion about the argument's validity, and the other ('both') indicating 'too much' (contradictory) evidence, i.e., the existence of both supportive and opposing arguments concerning the argument under consideration.

Both of these generalized approaches are a conservative extension of the standard approaches of giving semantics to abstract argumentation systems, in the sense that they do not exclude standard extensions or labelings, but rather offer additional points of views to the state of affairs as depicted by the argumentation framework. This allows us to introduce a brand new family of semantics that accommodate conflicts in the sense that internal attacks among accepted arguments are allowed, while the set of accepted arguments is not trivialized (i.e., it is not the case that every argument is necessarily accepted).

We introduce an extended set of criteria for selecting the most plausible four-valued labelings for an argumentation framework. These criteria are then justified by showing that the one-to-one relationship between extensions and labelings obtained for conflict-free semantics (see [7]) is carried on to a similar correspondence between the extended approaches for providing conflict-tolerant (paraconsistent) semantics. This also shows that in the case of conflict-tolerant semantics as well, extensions and labelings are each other's dual.

2 Preliminaries

Let us first recall some basic definitions and useful notions regarding abstract argumentation theory.

Definition 1. A (finite) *argumentation framework* [8] is a pair $\mathcal{AF} = \langle Args, att \rangle$, where *Args* is a finite set, the elements of which are called *arguments*, and *att* is a binary relation on $Args \times Args$ whose instances are called *attacks*. When $(A, B) \in att$ we say that *A attacks B* (or that *B* is *attacked by A*).

Given an argumentation framework $\mathcal{AF} = \langle Args, att \rangle$, in the sequel we shall use the following notations for an argument $A \in Args$ and a set of arguments $\mathcal{S} \subseteq Args$:

- The set of arguments that are *attacked by* A is $A^+ = \{B \in \mathit{Args} \mid \mathit{att}(A, B)\}$.
- The set of arguments that *attack* A is $A^- = \{B \in \mathit{Args} \mid \mathit{att}(B, A)\}$.

Similarly, $\mathcal{S}^+ = \bigcup_{A \in \mathcal{S}} A^+$ and $\mathcal{S}^- = \bigcup_{A \in \mathcal{S}} A^-$ denote, respectively, 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, we denote:

- The set of arguments that are *defended by* $\mathcal{S}$: $\mathrm{Def}(\mathcal{S}) = \{A \in \mathit{Args} \mid A^- \subseteq \mathcal{S}^+\}$.

Thus, an argument A is defended by $\mathcal{S}$ if each attacker of A is attacked by (an argument in) $\mathcal{S}$. The two primary principles of acceptable sets of arguments are now defined as follows:

Definition 2. Let $\mathcal{AF} = \langle \mathit{Args}, \mathit{att} \rangle$ be an argumentation framework.

- A set $\mathcal{S} \subseteq \mathit{Args}$ is *conflict-free* (with respect to $\mathcal{AF}$) iff $\mathcal{S} \cap \mathcal{S}^+ = \emptyset$.
- A conflict-free set $\mathcal{S} \subseteq \mathit{Args}$ is *admissible* for $\mathcal{AF}$, iff $\mathcal{S} \subseteq \mathrm{Def}(\mathcal{S})$.

Conflict-freeness assures that no argument in the set is attacked by another argument in the set, and admissibility guarantees, in addition, that the set is self defendant. A stronger notion is the following:

- A conflict-free set $\mathcal{S} \subseteq \mathit{Args}$ is *complete* for $\mathcal{AF}$, iff $\mathcal{S} = \mathrm{Def}(\mathcal{S})$.

The principles defined above are a cornerstone of a variety of extension-based semantics for an argumentation framework $\mathcal{AF}$, i.e., formalizations of sets of arguments that can collectively be accepted in $\mathcal{AF}$ (see, e.g., [8, 12]). In what follows, we shall usually denote an extension by Ext. This includes, among others, *grounded extensions* (the minimal set, with respect to set inclusion, that is complete for $\mathcal{AF}$), *preferred extensions* (the maximal subset of Args that is complete for $\mathcal{AF}$), *stable extensions* (any complete subset Ext of Args for which $\mathit{Ext}^+ = \mathit{Args} \backslash \mathit{Ext}$), and so forth.[1]

An alternative way to describe argumentation semantics is based on the concept of an *argument labeling* [5, 7]. The main definitions and the relevant results concerning this approach are surveyed below.

Definition 3. Let $\mathcal{AF} = \langle \mathit{Args}, \mathit{att} \rangle$ be an argumentation framework. An *argument labeling* is a complete function $\mathit{lab} : \mathit{Args} \to \{\mathsf{in}, \mathsf{out}, \mathsf{undec}\}$. We shall sometimes write $\mathsf{In}(\mathit{lab})$ for $\{A \in \mathit{Args} \mid \mathit{lab}(A) = \mathsf{in}\}$, $\mathsf{Out}(\mathit{lab})$ for $\{A \in \mathit{Args} \mid \mathit{lab}(A) = \mathsf{out}\}$ and $\mathsf{Undec}(\mathit{lab})$ for $\{A \in \mathit{Args} \mid \mathit{lab}(A) = \mathsf{undec}\}$.

In essence, an argument labeling expresses a position on which arguments one accepts (labeled in), which arguments one rejects (labeled out), and which arguments one abstains from having an explicit opinion about (labeled undec). Since a labeling lab of $\mathcal{AF} = \langle \mathit{Args}, \mathit{att} \rangle$ can be seen as a partition of Args, we shall sometimes write it as a triple $\langle \mathsf{In}(\mathit{lab}), \mathsf{Out}(\mathit{lab}), \mathsf{Undec}(\mathit{lab}) \rangle$.

[1] Common definitions of conflict-free extension-based semantics for argumentation frameworks, different methods for computing them, and computational complexity analysis appear, e.g., in [1, 6, 8, 9, 10, 11].

Definition 4. Consider the following conditions on a labeling *lab* and an argument A in a framework $\mathcal{AF} = \langle Args, att \rangle$:

Pos1 If $lab(A) = \mathsf{in}$, there is no $B \in A^-$ such that $lab(B) = \mathsf{in}$.
Pos2 If $lab(A) = \mathsf{in}$, for every $B \in A^-$ it holds that $lab(B) = \mathsf{out}$.
Neg If $lab(A) = \mathsf{out}$, there exists some $B \in A^-$ such that $lab(B) = \mathsf{in}$.
Neither If $lab(A) = \mathsf{undec}$, not for every $B \in A^-$ it holds that $lab(B) = \mathsf{out}$ and there does not exist a $B \in A^-$ such that $lab(B) = \mathsf{in}$.

Given a labeling *lab* of an argumentation framework $\langle Args, att \rangle$, we say that

- *lab* is *conflict-free* if for every $A \in Args$ it satisfies conditions **Pos1** and **Neg**,
- *lab* is *admissible* if for every $A \in Args$ it satisfies conditions **Pos2** and **Neg**,
- *lab* is *complete* if it is admissible and for every $A \in Args$ it satisfies **Neither**.

Again, the labelings considered above serve as a basis for a variety of labeling-based semantics that have been proposed for an argumentation framework $\mathcal{AF}$, each one of which is a counterpart of a corresponding extension-based semantics. This includes, for instance, the *grounded labeling* (a complete labeling for $\mathcal{AF}$ with a minimal set of in-assignments), the *preferred labeling* (a complete labeling for $\mathcal{AF}$ with a maximal set of in-assignments), *stable labelings* (any complete labeling of $\mathcal{AF}$ without undec-assignments), and so forth.

The next correspondence between extensions and labelings is shown in [7]:

Proposition 1. *Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework, $\mathcal{CFE}$ the set of all conflict-free extensions of $\mathcal{AF}$, and $\mathcal{CFL}$ the set of all conflict-free labelings of $\mathcal{AF}$. Consider the function $\mathcal{LE}_{\mathcal{AF}} : \mathcal{CFL} \to \mathcal{CFE}$, defined by $\mathcal{LE}_{\mathcal{AF}}(lab) = \mathsf{In}(lab)$ and the function $\mathcal{EL}_{\mathcal{AF}} : \mathcal{CFE} \to \mathcal{CFL}$, defined by $\mathcal{EL}_{\mathcal{AF}}(Ext) = \langle Ext, Ext^+, Args \setminus (Ext \cup Ext^+) \rangle$. It holds that:*

1. *If Ext is an admissible (respectively, complete) extension, then $\mathcal{EL}_{\mathcal{AF}}(Ext)$ is an admissible (respectively, complete) labeling.*
2. *If lab is an admissible (respectively, complete) labeling, then $\mathcal{LE}_{\mathcal{AF}}(lab)$ is an admissible (respectively, complete) extension.*
3. *When the domain and range of $\mathcal{EL}_{\mathcal{AF}}$ and $\mathcal{LE}_{\mathcal{AF}}$ are restricted to complete extensions and complete labelings of $\mathcal{AF}$, then these functions become bijections and each other's inverses, making complete extensions and complete labelings one-to-one related.*

3 Conflicts Tolerance

In this section we extend the two approaches considered previously in order to define conflict-tolerant semantics for abstract argumentation frameworks. Recall that our purpose here is twofold:

1. Introducing self-referring argumentation and avoiding information loss that may be caused by the conflict-freeness requirement (thus, for instance, it may be better to accept extensions with a small fragment of conflicting arguments than, say, sticking to the empty extension).

2. Refining the undec-indication in standard labeling systems, which reflects (at least) two totally different situations: One case is that the reasoner abstains from having an opinion about an argument because there are no indications whether this argument should be accepted or rejected. Another case that may cause a neutral opinion is that there are simultaneous considerations for and against accepting a certain argument. These two cases should be distinguishable, since their outcomes may be different.

3.1 Four-Valued Paraconsistent Labelings

Item 2 above may serve as a motivation for the following definition:

Definition 5. Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework. A *four-valued labeling* for $\mathcal{AF}$ is a complete function $lab : Args \to \{\mathsf{in}, \mathsf{out}, \mathsf{none}, \mathsf{both}\}$. We shall sometimes write $\mathsf{None}(lab)$ for $\{A \in Args \mid lab(A) = \mathsf{none}\}$ and $\mathsf{Both}(lab)$ for $\{A \in Args \mid lab(A) = \mathsf{both}\}$.

As before, the labeling function reflects the state of mind of the reasoner regarding each argument in $\mathcal{AF}$. The difference is, of-course, that four-valued labelings are a refinement of 'standard' labelings (in the sense of Definition 3), so that *four* states are allowed. Thus, we continue to denote by $\mathsf{In}(lab)$ the set of arguments that one accepts and by $\mathsf{Out}(lab)$ the set of arguments that one rejects, but now the set $\mathsf{Undec}(lab)$ is splitted to two new sets: $\mathsf{None}(lab)$, consisting of arguments that may neither be accepted nor rejected, and $\mathsf{Both}(lab)$, consisting of arguments who have both supportive and rejective evidences. Since a four-valued labeling lab is a partition of $Args$, we sometimes write it as a quadruple $\langle \mathsf{In}(lab), \mathsf{Out}(lab), \mathsf{None}(lab), \mathsf{Both}(lab) \rangle$.

Definition 6. 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 four-valued labeling $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ of $\mathcal{AF}$,[2] defined for every $A \in Args$ as follows:

$$p\mathcal{EL}_{\mathcal{AF}}(Ext)(A) = \begin{cases} \mathsf{in} & \text{if } A \in Ext \text{ and } A \notin Ext^+, \\ \mathsf{both} & \text{if } A \in Ext \text{ and } A \in Ext^+, \\ \mathsf{out} & \text{if } A \notin Ext \text{ and } A \in Ext^+, \\ \mathsf{none} & \text{if } A \notin Ext \text{ and } A \notin Ext^+. \end{cases}$$

  A four-valued labeling that is induced by some subset of $Args$ is called a *paraconsistent labeling* (or a *p-labeling*) of $\mathcal{AF}$.
- Given a four-valued labeling lab of $\mathcal{AF}$, the set of arguments that is *induced by* (or, is *associated with*) lab is defined by

$$p\mathcal{LE}_{\mathcal{AF}}(lab) \;=\; \mathsf{In}(lab) \;\cup\; \mathsf{Both}(lab).$$

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

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 in or both) should be included in the extension (even if there are also opposing indications). The transformation from an extension Ext to its induced labeling function $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is motivated by the aspiration to accept the arguments in the extension by marking them as either in or both. 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.

Definition 6 indicates a one-to-one correspondence between sets of arguments of an argumentation framework and the labelings that are induced by them. It follows that while there are $4^{|Args|}$ four-valued labelings for an argumentation framework $\mathcal{AF} = \langle Args, att\rangle$, the number of paraconsistent labelings (p-labelings) for $\mathcal{AF}$ is limited by the number of the subsets of $Args$, i.e., $2^{|Args|}$.

Example 1. Consider the argumentation framework $\mathcal{AF}_1$ of Figure 1.

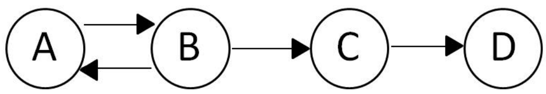

Fig. 1. The argumentation framework $\mathcal{AF}_1$

To compute the paraconsistent labelings of $\mathcal{AF}_1$, note for instance that if for some $Ext \subseteq Args$ it holds that $p\mathcal{EL}_{\mathcal{AF}}(Ext)(A) = \mathsf{in}$, then $A \in Ext$ and $A \notin Ext^+$, which implies, respectively, that $B \in Ext^+$ and $B \notin Ext$, thus B must be labeled out. Similarly, if A is labeled out then B must be labeled in, if A is labeled both, B must be labeled both as well, and if A is labeled none, so B is labeled none. These labelings correspond to the four possible choices of either accepting exactly one of the mutually attacking arguments A and B, accepting both of them, or rejecting both of them. In turn, each such choice is augmented with four respective options for labeling C and D. Table 1 lists the corresponding sixteen p-labelings of $\mathcal{AF}_1$.

A p-labeling may be regarded as a description of the state of affairs for *any* chosen set of arguments in a framework. For instance, the second p-labeling in Table 1 (Example 1) indicates that if $\{A, C, D\}$ is the accepted set of arguments, then B is rejected (labeled out) since it is attacked by an accepted argument, and the status of D is ambiguous (so it is labeled both), since on one hand it is included in the set of accepted arguments, but on the other hand it is attacked by an accepted argument. Note, further, that choosing D as an accepted argument in this case is somewhat arguable, since D is not defended by the set $\{A, C, D\}$.

The discussion above implies that the role of a p-labeling is *indicative* rather than *justificatory*; A labeling that is induced by Ext describes the role of each argument in the framework according to Ext, but it does not *justify* the choice

	A	B	C	D	Induced set
1	in	out	in	out	$\{A, C\}$
2	in	out	in	both	$\{A, C, D\}$
3	in	out	none	in	$\{A, D\}$
4	in	out	none	none	$\{A\}$
5	out	in	out	in	$\{B, D\}$
6	out	in	out	none	$\{B\}$
7	out	in	both	out	$\{B, C\}$
8	out	in	both	both	$\{B, C, D\}$

	A	B	C	D	Induced set
9	none	none	in	out	$\{C\}$
10	none	none	in	both	$\{C, D\}$
11	none	none	none	in	$\{D\}$
12	none	none	none	none	$\{\}$
13	both	both	out	in	$\{A, B, D\}$
14	both	both	out	none	$\{A, B\}$
15	both	both	both	out	$\{A, B, C\}$
16	both	both	both	both	$\{A, B, C, D\}$

Table 1. The p-labelings of $\mathcal{AF}_1$

of *Ext* as a plausible extension for the framework. For the latter, we should pose further restrictions on the p-labelings. This is what we do next.

Definition 7. Given an argumentation framework $\mathcal{AF} = \langle Args, att \rangle$, a p-labeling *lab* for $\mathcal{AF}$ is called *p-admissible*, if it satisfies the following rules:

pIn If $lab(A) = \mathsf{in}$, for every $B \in A^-$ it holds that $lab(B) = \mathsf{out}$.
pOut If $lab(A) = \mathsf{out}$, there exists $B \in A^-$ such that $lab(B) \in \{\mathsf{in}, \mathsf{both}\}$.
pBoth If $lab(A) = \mathsf{both}$, for every $B \in A^-$ it holds that $lab(B) \in \{\mathsf{out}, \mathsf{both}\}$ and there exists some $B \in A^-$ such that $lab(B) = \mathsf{both}$.
pNone If $lab(A) = \mathsf{none}$, for every $B \in A^-$ it holds that $lab(B) \in \{\mathsf{out}, \mathsf{none}\}$.

The constraints in Definition 7 may be strengthen as follows:

Definition 8. Given an argumentation framework $\mathcal{AF} = \langle Args, att \rangle$, a p-labeling *lab* for $\mathcal{AF}$ is called *p-complete*, if it satisfies the following rules:

pIn$^+$ $lab(A) = \mathsf{in}$ iff for every $B \in A^-$ it holds that $lab(B) = \mathsf{out}$.
pOut$^+$ $lab(A) = \mathsf{out}$ iff there is $B \in A^-$ such that $lab(B) \in \{\mathsf{in}, \mathsf{both}\}$ and there is some $B \in A^-$ such that $lab(B) \in \{\mathsf{in}, \mathsf{none}\}$.
pBoth$^+$ $lab(A) = \mathsf{both}$ iff for every $B \in A^-$ it holds that $lab(B) \in \{\mathsf{out}, \mathsf{both}\}$ and there exists some $B \in A^-$ such that $lab(B) = \mathsf{both}$.
pNone$^+$ $lab(A) = \mathsf{none}$ iff for every $B \in A^-$ it holds that $lab(B) \in \{\mathsf{out}, \mathsf{none}\}$ and there exists some $B \in A^-$ such that $lab(B) = \mathsf{none}$.

Example 2. Consider again the p-labelings for $\mathcal{AF}_1$ (Example 1), listed in Table 1.

- The rule **pIn** is violated by labelings 3, 9, 10, 11, and the rule **pBoth** is violated by labelings 2, 7, 8, 10. Therefore, the p-admissible labelings in this case are 1, 4, 5, 6, 12–16.
- Among the p-admissible labelings in the previous item, labelings 4 and 6 violate **pNone**$^+$, and labelings 13–15 violate **pOut**$^+$. Thus, the p-complete labelings of $\mathcal{AF}_1$ are 1, 5, 12 and 16.[3]

[3] Intuitively, these labelings represent the most plausible states corresponding to the four possible choices of arguments among the mutually attacking A and B.

In Section 3.3 and Section 4 we shall justify the rules in Definitions 7 and 8 by showing the correspondence between p-admissible/p-complete labelings and related extensions.

3.2 Paraconsistent Extensions

Recall that Item 1 at the beginning of Section 3 suggests that the 'conflict-freeness' requirement in Definition 2 may be lifted. However, the other properties in the same definition, implying that an argument in an extension must be defended, are still necessary.

Definition 9. Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework and let $Ext \subseteq Args$. We say that 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 $Ext = \mathrm{Def}(Ext)$.

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.

It is well-known that every argumentation framework has at least one complete extension. However, there are cases (see, e.g., the argumentation framework $\mathcal{AF}_2$ in Figure 2) that the only complete extension for a framework is the empty set. The next proposition shows that this is not the case regarding p-complete extensions.

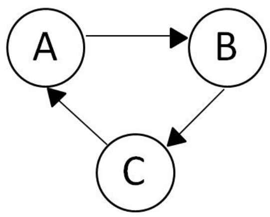

Fig. 2. The argumentation framework $\mathcal{AF}_2$

Proposition 2. *Any argumentation framework has a nonempty p-complete extension.*[4]

Example 3. 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 of $\mathcal{AF}_2$), and $\{A, B, C\}$.

[4] Due to lack of space proofs are omitted.

3.3 Relating Paraconsistent Extensions and Labelings

We are now ready to consider the extension-based semantics induced by paraconsistent labelings. We show, in particular, that as in the case of (conflict-free) complete labelings and (conflict-free) complete extensions, there is a one-to-one correspondence between them, thus they represent two equivalent approaches for giving conflict-tolerant semantics to abstract argumentation frameworks.

Proposition 3. *If Ext is a p-admissible extension of $\mathcal{AF}$ then $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a p-admissible labeling of $\mathcal{AF}$.*

Proposition 4. *If lab is a p-admissible labeling of $\mathcal{AF}$ then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is a p-admissible extension for $\mathcal{AF}$.*

Proposition 5. *Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework.*

- *For every p-admissible labeling lab for $\mathcal{AF}$, $p\mathcal{EL}_{\mathcal{AF}}(p\mathcal{LE}_{\mathcal{AF}}(lab)) = lab$.*
- *For every p-admissible extension Ext of $\mathcal{AF}$, $p\mathcal{LE}_{\mathcal{AF}}(p\mathcal{EL}_{\mathcal{AF}}(Ext)) = Ext$.*

By Propositions 3, 4, and 5, we conclude the following:

Corollary 1. *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.*

It follows that p-admissible extensions and p-admissible labelings are, in essence, different ways of describing the same thing (see also Figure 3 below).

Note 1. In a way, the correspondence between p-admissible extensions and p-admissible labelings of an argumentation framework is tighter than the correspondence between (conflict-free) admissible extensions and (conflict-free) admissible labelings, as depicted in [7] (see Section 2). Indeed, as indicated in [7], admissible labelings and admissible extensions have a many-to-one relationship: each admissible labeling is associated with exactly one admissible extension, but an admissible extension may be associated with several admissible labelings. For instance, in the argumentation framework $\mathcal{AF}_1$ of Figure 1 (Example 1), $lab_1 = \langle \{B\}, \{A, C\}, \{D\} \rangle$ and $lab_2 = \langle \{B\}, \{A\}, \{C, D\} \rangle$ are different admissible labelings that are associated with the same admissible extension $\{B\}$. Note that, viewed as *four-valued* labelings into $\{\mathsf{in}, \mathsf{out}, \mathsf{none}\}$, only lab_1 is p-admissible, since lab_2 violates **pNone**. Indeed, the p-admissible extension $\{B\}$ is associated with exactly *one* p-admissible labeling (number 6 in Table 1), as guaranteed by the last corollary.

Let us now consider p-complete labelings.

Proposition 6. *If Ext is a p-complete extension of $\mathcal{AF}$ then $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a p-complete labeling of $\mathcal{AF}$.*

Proposition 7. *If lab is a p-complete labeling of $\mathcal{AF}$ then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is a p-complete extension for $\mathcal{AF}$.*

Proposition 8. *Let $\mathcal{AF}$ be an argumentation framework.*

– *For every p-complete labeling lab for $\mathcal{AF}$, $p\mathcal{EL}_{\mathcal{AF}}(p\mathcal{LE}_{\mathcal{AF}}(lab)) = lab$.*
– *For every p-complete extension Ext of $\mathcal{AF}$, $p\mathcal{LE}_{\mathcal{AF}}(p\mathcal{EL}_{\mathcal{AF}}(Ext)) = Ext$.*

By Propositions 6, 7, and 8, we conclude the following:

Corollary 2. *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.*

It follows that p-complete extensions and p-complete labelings are different ways of describing the same thing (see also Figure 3). This is in correlation with the results for conflict-free semantics, according to which there is a one-to-one relationship between complete extensions and complete labelings (Proposition 1).

Example 4. Consider again the framework $\mathcal{AF}_1$ of Example 1.

1. By Example 2 and Propositions 4, the p-admissible extensions of $\mathcal{AF}_1$ are induced by labelings 1, 4, 5, 6, 12–16 in Table 1, i.e., $\{A, C\}$, $\{A\}$, $\{B, D\}$, $\{B\}$, $\emptyset$, $\{A, B, D\}$, $\{A, B\}$, $\{A, B, C\}$, and $\{A, B, C, D\}$ (respectively).
2. By Example 2 and Propositions 7, the p-complete extensions of $\mathcal{AF}_1$ are those that are induced by labelings 1, 5, 12 and 16 in Table 1, namely $\{A, C\}$, $\{B, D\}$, $\emptyset$, and $\{A, B, C, D\}$ (respectively).

4 From Conflict-Tolerant to Conflict-Free Semantics

In this section we show that the variety of 'standard' semantics for argumentation frameworks, based on conflict-free extensions and conflict-free labeling functions, are still available in our conflict-tolerant setting. First, we consider admissible extensions (Definition 2) and admissible labelings (Definition 4).

Proposition 9. *Let lab be a p-admissible labeling for an argumentation framework $\mathcal{AF}$. If lab is into $\{\mathsf{in}, \mathsf{out}, \mathsf{none}\}$,*[5] *then it is admissible.*

As Note 1 shows, the converse of the last proposition does not hold. Indeed, as indicated by Caminada and Gabbay [7], there is a many-to-one relationship between admissible labelings and admissible extensions. On the other hand, by the next proposition (together with Corollary 2), there is a one-to-one relationship between both-free p-admissible labelings and admissible extensions.

Proposition 10. *Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework. Then*

1. *If lab is a* both*-free p-admissible labeling for $\mathcal{AF}$, then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is an admissible extension of $\mathcal{AF}$.*
2. *If Ext is an admissible extension of $\mathcal{AF}$ then $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a* both*-free p-admissible labeling for $\mathcal{AF}$.*

[5] In which case *lab* is called 'both-free'.

Let us now consider complete extensions and complete labelings. The next two propositions are the analogue, for complete labelings and complete extensions, of Propositions 9 and 10:

Proposition 11. *Let lab be a p-complete labeling for an argumentation framework* $\mathcal{AF}$*. If lab is into* $\{\mathsf{in}, \mathsf{out}, \mathsf{none}\}$*, then it is complete.*

Proposition 12. *Let* $\mathcal{AF} = \langle Args, att \rangle$ *be an argumentation framework. Then*

1. *If lab is a* both*-free p-complete labeling for* $\mathcal{AF}$*, then* $p\mathcal{LE}_{\mathcal{AF}}(lab)$ *is a complete extension of* $\mathcal{AF}$*.*
2. *If Ext is a complete extension of* $\mathcal{AF}$ *then* $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ *is a* both*-free p-complete labeling for* $\mathcal{AF}$*.*

Proposition 13. *Let* $\mathcal{AF}$ *be an argumentation framework. Then lab is a complete labeling for* $\mathcal{AF}$ *iff it is a* both*-free p-complete labeling for* $\mathcal{AF}$*.*

Figure 3 summarizes the relations between the conflict-free semantics and the conflict-tolerant semantics considered so far (the arrows in the figure denote "is-a" relationships, and the double-arrows denote one-to-one relationships).

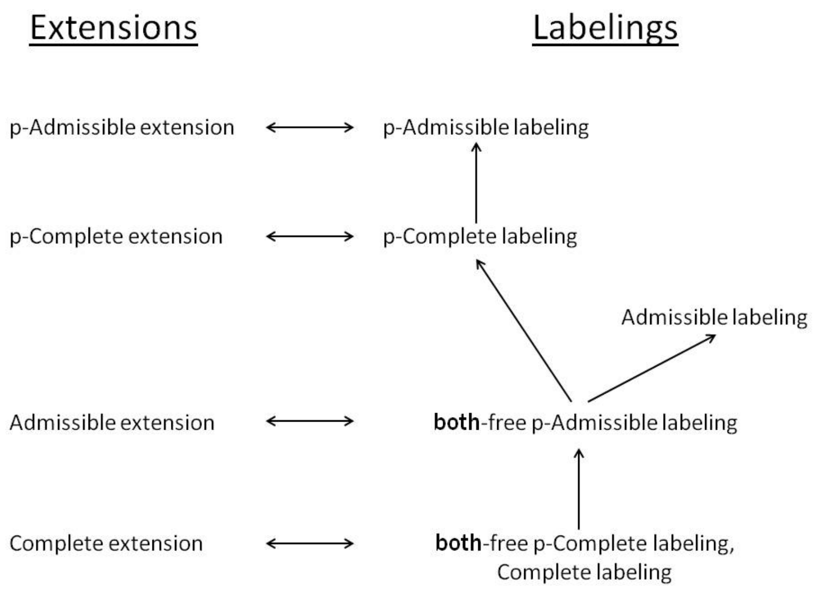

Fig. 3. Conflict-free and conflict-tolerant semantics

By Proposition 13, a variety of conflict-free, extension-based (Dung-style) semantics for abstract argumentation frameworks may be defined in terms of both-free p-complete labelings. For instance,

- Ext is a grounded extension of $\mathcal{AF}$ iff it is induced by a both-free p-complete labeling lab of $\mathcal{AF}$ such that there is no both-free p-complete labeling lab' of $\mathcal{AF}$ with $\mathsf{In}(lab') \subset \mathsf{In}(lab)$.
- Ext is a preferred extension of $\mathcal{AF}$ iff it is induced by a both-free p-complete labeling lab of $\mathcal{AF}$ such that there is no both-free p-complete labeling lab' of $\mathcal{AF}$ with $\mathsf{In}(lab) \subset \mathsf{In}(lab')$.
- Ext is a semi-stable extension of $\mathcal{AF}$ iff it is induced by a both-free p-complete labeling lab of $\mathcal{AF}$ such that there is no both-free p-complete labeling lab' of $\mathcal{AF}$ with $\mathsf{None}(lab') \subset \mathsf{None}(lab)$.
- Ext is a stable extension of $\mathcal{AF}$ iff it is induced by a both-free p-complete labeling lab of $\mathcal{AF}$ such that $\mathsf{None}(lab) = \emptyset$.

By the last item, stable extensions correspond to $\{\mathsf{both}, \mathsf{none}\}$-free p-complete labelings:

Corollary 3. *Let $\mathcal{AF} = \langle Args, att \rangle$ be an argumentation framework.*

1. *If lab is a $\{\mathsf{both}, \mathsf{none}\}$-free p-complete labeling for $\mathcal{AF}$, then $p\mathcal{LE}_{\mathcal{AF}}(lab)$ is a stable extension of $\mathcal{AF}$.*
2. *If Ext is a stable extension of $\mathcal{AF}$, then $p\mathcal{EL}_{\mathcal{AF}}(Ext)$ is a $\{\mathsf{both}, \mathsf{none}\}$-free p-complete labeling for $\mathcal{AF}$.*

Example 5. Consider again the framework $\mathcal{AF}_1$ of Example 1.

1. By Proposition 12, the complete extensions of $\mathcal{AF}_1$ are induced by the both-free p-complete labelings, i.e., $\{A, C\}$, $\{B, D\}$ and $\emptyset$ (which are the both-free labelings among those mentioned in Items 2 of Examples 2 and 4).
2. By Corollary 3, the stable extensions of $\mathcal{AF}_1$ are those induced by the none-free labelings among the labeling in the previous item, i.e., $\{A, C\}$ and $\{B, D\}$.

5 Conclusion

We have introduced a four-valued approach to provide conflict-tolerant semantics for abstract argumentation frameworks. Such an approach may be beneficial for several reasons:

- From a purely theoretical point of view, we have shown that the correlation between the labeling-based approach and the extension-based approach to argumentation theory is preserved also when conflict-freeness is abandon. Interestingly, as indicated in Note 1, in our framework this correlation holds also between admissibility-based labelings and admissibility-based extensions, which is *not* the case in the conflict-free setting of [7].
- From a more pragmatic point of view, new types of semantics are introduced, which accommodate conflicts, yet they are not trivialized by inconsistency. It is shown that this setting is not a substitute of standard (conflict-free) semantics, but rather a generalized framework, offering an option for inter-attacks when such attacks make sense or are unavoidable.

– Conflicts handling in argumentation systems turns out to be more evasive than what it looks like at first sight. In fact, conflicts may implicitly arise even in conflict-free semantics, because such semantics simulate binary attacks and not collective conflicts (this is demonstrated in the last example of [2]). In this respect, the possibility of having conflicts is not completely ruled out even in some conflict-free semantics (such as CF2 and stage semantics; see [2]), and our approach may be viewed as an explication of this possibility.

Our setting may be related to other settings that to the best of our knowledge have not been connected so far to argumentation theory. For instance, the resemblance to Belnap's well-known four-valued framework for computerized reasoning [4] is evident. Moreover, the use of four-valued labelings suggests that the four-valued signed systems, used in [1] for representing conflict-free semantics of argumentation frameworks, may be incorporated for representing the conflict-tolerant semantics in this paper. We leave this for a future work.

Acknowledgement Martin Caminada is thanked for useful discussions on topics related to this paper.

References

[1] O. Arieli and M. W. A. Caminada. A general QBF-based framework for formalizing argumentation. In *Proc. COMMA'12*. IOS Press, 2012. In press.

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

[3] P. Baroni and M. Giacomin. Semantics for abstract argumentation systems. In Rahwan and Simary [12], pages 25–44.

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

[5] M. W. A. Caminada. On the issue of reinstatement in argumentation. In *Proc. JELIA'06*, LNAI 4160, pages 111–123. Springer, 2006.

[6] M. W. A. Caminada, W. A. Carnielli, and P. Dunne. Semi-stable semantics. *Journal of Logic and Computation*, 2012. In print.

[7] M. W. A. Caminada and D. M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2-3):109–145, 2009.

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

[9] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10–15):642–674, 2007.

[10] W. Dvořák. On the complexity of computing the justification status of an argument. In *Proc. TAFA-11*, LNAI 7132, pages 32–49. Springer, 2011.

[11] U. Egly, S. A. Gaggl, and S. Woltran. Answer-set programming encodings for argumentation frameworks. *Argument and Computation*, 1(2):144–177, 2010.

[12] I. Rahwan and G. R. Simari. *Argumentation in Artificial Intelligence*. Springer, 2009.