Instantiating rule-based defeasible theories in abstract dialectical frameworks and beyond

Hannes Strass*

Computer Science Institute, Leipzig University
Augustusplatz 10, 04109 Leipzig, Germany

Abstract

We present a translation from defeasible theory bases to abstract dialectical frameworks, a recent generalisation of abstract argumentation frameworks. Using several problematic examples from the literature, we first show how our translation addresses important issues of existing approaches. We then prove that the translated frameworks satisfy the rationality postulates closure and direct/indirect consistency. Furthermore, the frameworks can detect inconsistencies in the set of strict inference rules and cyclic (strict and defeasible) supports amongst literals. We also show that the translation involves at most a quadratic blowup and is therefore effectively and efficiently computable. In the last part of the paper, we also define a direct, possible-worlds semantics for defeasible theory bases, which illustrates the technical difficulties arising in this process. In particular, the possible-worlds semantics is eager to apply defeasible rules, which is in contrast to the previously studied translation-based approaches.

Keywords: abstract dialectical frameworks, abstract argumentation frameworks, defeasible theories, rule bases

1. Introduction

Abstract argumentation frameworks (AFs) [11] are a formalism that is widely used in argumentation research. Such an AF consists of a set of arguments and an attack relation between these arguments. Their semantics determines which sets of arguments of a given AF can be accepted according to specific criteria. A common way to employ Dung's AFs is as abstraction formalism. In this view, expressive languages are used to model concrete argumentation scenarios, and translations into Dung AFs provide these original languages with semantics. The advantage of translating into an argumentation formalism is

*Corresponding author
Email address: strass@informatik.uni-leipzig.de (Hannes Strass)

that the resulting semantics can be given a dialectical interpretation, which can be used to inform humans how a particular conclusion was inferred.

However, the approach is not without its problems. Caminada and Amgoud [6] reported some difficulties they encountered when defining an abstract argumentation-based semantics for defeasible theory bases. Defeasible theory bases are simple logic-inspired formalisms working with inference rules on a set of literals. Inference rules can be strict, in which case the conclusion of the inference (a literal) must necessarily hold whenever all antecedents (also literals) hold. Inference rules can also be defeasible, which means that the conclusion *usually* holds whenever the antecedents hold. Here, the word "usually" suggests that there could be exceptional cases where a defeasible rule has not been applied [17].

In response to the problems they encountered, Caminada and Amgoud [6] stated general rationality postulates for AFs based on defeasible theories. The intention of these postulates is to mathematically capture what humans perceive as rational behaviour from the semantics of defeasible theory bases. First of all the *closure* postulate says that whatever model or extension the target formalism (the AF) produces, it must be closed under application of strict rules, meaning that all applicable strict rules have been applied. Direct and indirect *consistency* postulates express that any model or extension of the target formalism must be internally consistent with respect to the literals of the defeasible theory base (directly) and even with respect to application of strict rules (indirectly).

Later, Wyner et al. [23] criticised Caminada and Amgoud's definition of arguments on ontological grounds and gave an alternative translation. We are agnostic with respect to Wyner et al.'s criticism, but use their translation as a starting point for our own work. Such a further refinement is necessary since the translation of Wyner et al. [23] still yields unintuitive results on benchmark examples and does not satisfy the closure and indirect consistency postulates. Wyner et al. [24] later fixed these issues by adding a meta-level integrity constraint on the obtained extensions, thus ruling out violation of the postulates. Our translation has this integrity constraint built into it, such that models can be taken as they are.

The basis of our solution to the aforementioned problems is a shift in the target language. While until now abstract argumentation frameworks were the formalism of choice, we will use the more general abstract *dialectical* frameworks (ADFs) [4]. Where AFs allow only attacks between arguments, ADFs can also represent support relations and many more. More specifically, in an AF an argument is accepted if none of its attackers is accepted. The same can be expressed in an ADF. But ADFs can also express that an argument is only accepted if all of its supporters are accepted, or the argument is accepted if *some* of its supporters are accepted, or it is accepted if some *attacker* is *not* accepted, and many more.

The modelling capacities of ADFs in comparison to AFs – which we studied previously [19, 5] – enables us to give a direct and straightforward translation from defeasible theory bases to abstract dialectical frameworks. We will show that this translation – the first main contribution of this paper – treats the

benchmark examples right and satisfies the rationality postulates of Caminada and Amgoud [6]. We consider this further important evidence that abstract dialectical frameworks are useful tools for representing and reasoning about argumentation scenarios. We also perform a complexity analysis of our translation; this is significant in that we are not aware of complexity analyses of the mentioned previous approaches.

The availability of support in ADFs (in contrast to AFs) as a target formalism will be of fundamental importance to our translation. Among other things, it will allow us to resolve cyclic dependencies among literals in a defeasible theory base in a straightforward way. The treatment of such support cycles is built into ADF standard semantics, which can be considered a product of decades of research into nonmonotonic knowledge representation languages.

As our second main contribution, we introduce a possible-worlds semantics for defeasible theory bases. This provides a language for formulating different intuitions about the meaning of strict and defeasible rules. Furthermore, it nicely illustrates the difficulties in formally defining semantics for collections of such rules. The semantics is inspired by possible-worlds semantics for autoepistemic logic [10], we therefore indirectly present potential epistemic modal readings of strict and defeasible rules.

In the rest of the paper, we first recall the necessary background on defeasible theory bases, abstract argumentation frameworks and abstract dialectical frameworks. In Section 3 we look at the translations of Caminada and Amgoud [6] and Wyner et al. [23], discuss some problems of these, and introduce generalised versions of the rationality postulates. In Section 4 we then define our own translation. We show how it treats the problematic examples, prove that it satisfies the (generalised versions of the) rationality postulates and analyse its computational complexity. We then introduce our direct semantics for defeasible theories and illustrate its behaviour on several examples, and afterwards clarify its connections to autoepistemic logic. We conclude with a discussion of related and future work. This paper is a revised and extended version of [20].

2. Background

Defeasible Theories. Following Caminada and Amgoud [6], we use a set Lit of literals that are built using syntactical negation $\neg\cdot$ and define a semantic negation function $\overline{\cdot}$ such that for an atom p we have $\overline{p} = \neg p$ and $\overline{\neg p} = p$. Throughout the paper, we assume that Lit is closed under negation in the sense that $\psi \in Lit$ implies $\overline{\psi} \in Lit$. A set $S \subseteq Lit$ of literals is *consistent* iff there is no literal $\psi \in Lit$ such that both $\psi \in S$ and $\neg\psi \in S$. For literals $\phi_1, \ldots, \phi_n, \psi \in Lit$, a *strict rule* over Lit is of the form $r : \phi_1, \ldots, \phi_n \rightarrow \psi$; a *defeasible rule* over Lit is of the form $r : \phi_1, \ldots, \phi_n \Rightarrow \psi$. (The only difference is the arrows.) Here r is the unique *rule name*, the literals $\phi_1, \ldots, \phi_n$ constitute the *rule body* and ψ is the *rule head* or *conclusion*. Intuitively, a strict rule says that the rule head is necessarily true whenever all body literals are true; a defeasible rule says that the head ψ is *usually* true whenever all body literals are true. In definitions, we use the symbol $\Rrightarrow$ as meta-level variable for $\rightarrow$ and $\Rightarrow$.

For a set $M \subseteq Lit$ of literals and a set *StrInf* of strict rules over *Lit*, we say that M is *closed under StrInf* iff $r : \phi_1, \ldots, \phi_n \to \psi \in StrInf$ and $\phi_1, \ldots, \phi_n \in M$ imply $\psi \in M$. Accordingly, the *closure of M under StrInf* is the smallest set $Cl_{StrInf}(M)$ that contains M and is closed under *StrInf*. A *defeasible theory* or *theory base* is a triple $(Lit, StrInf, DefInf)$ where *Lit* is a set of literals, *StrInf* is a set of strict rules over *Lit* and *DefInf* is a set of defeasible rules over *Lit*. The semantics of theory bases is usually defined via a translation to abstract argumentation frameworks, which will be introduced next.

Abstract Argumentation Frameworks. Dung [11] introduced argumentation frameworks as pairs $\Theta = (A, R)$ where A is a set and $R \subseteq A \times A$ a relation. The intended reading of an AF Θ is that the elements of A are arguments whose internal structure is abstracted away. The only information about the arguments is given by the relation R encoding a notion of attack: a pair $(a, b) \in R$ expresses that argument a attacks argument b in some sense.

The purpose of semantics for argumentation frameworks is to determine sets of arguments (called *extensions*) which are acceptable according to various standards. We will only be interested in so-called *stable* extensions, sets S of arguments that do not attack each other and attack all arguments not in the set. More formally, a set $S \subseteq A$ of arguments is *conflict-free* iff there are no $a, b \in S$ with $(a, b) \in R$. A set S is a *stable extension* for (A, R) iff it is conflict-free and for all $a \in A \setminus S$ there is a $b \in S$ with $(b, a) \in R$.

Abstract Dialectical Frameworks. Brewka and Woltran [4] introduced abstract dialectical frameworks as a powerful generalisation of Dung AFs that are able to capture not only attack and support, but also more general notions such as joint attack and joint support.

Definition 1. An *abstract dialectical framework* is a triple $\Xi = (S, L, C)$ where

- S is a set of *statements*,
- $L \subseteq S \times S$ is a set of *links*, where $par(s) \stackrel{\text{def}}{=} \{r \in S \mid (r, s) \in L\}$
- $C = \{C_s\}_{s \in S}$ is a set of total functions $C_s : 2^{par(s)} \to \{in, out\}$.

Intuitively, the function C_s for a statement s determines the acceptance status of s, which naturally depends on the status of its parent nodes. Any such function C_s can alternatively be represented by a propositional formula φ_s over the vocabulary $par(s)$. The understanding here is that for $M \subseteq par(s)$, $C_s(M) = in$ iff M is a model of φ_s (written $M \models \varphi_s$), where an interpretation is identified with the set of atoms that are evaluated to true.

Brewka and Woltran [4] introduced several semantical notions for ADFs. First, for an ADF $\Xi = (S, L, C)$ where C is given by a set of propositional formulas φ_s for each $s \in S$, a set $M \subseteq S$ is a *model for* Ξ iff for all statements s we have: $s \in M$ iff $M \models \varphi_s$.

Example 1 (Abstract dialectical framework). Consider the ADF $D = (S, L, C)$ with statements $S = \{a, b, c, d\}$, links $L = \{(a, c), (b, b), (b, c), (b, d)\}$ and acceptance functions given by the formulas $\varphi_a = \top$, $\varphi_b = b$, $\varphi_c = a \wedge b$ and $\varphi_d = \neg b$. Intuitively, these acceptance conditions express that (1) a is always accepted, (2) b supports itself, (3) c needs the joint support of a and b, and (4) d is attacked by b. The two models of D are $M_1 = \{a, b, c\}$ and $M_2 = \{a, d\}$.

The semantics of ADFs can be defined using operators [5]. In this paper, we are only interested in two-valued semantics, that is, models and stable models. The definition of the latter is based on the notion of a reduct and an operator originally introduced by Brewka and Woltran [4]. The operator Γ_Ξ takes two sets A, R of statements, where the intuition is that all statements in A are accepted and those in R are rejected. (So those in $S \setminus (A \cup R)$ are undecided.) According to these acceptance statuses, the operator evaluates all acceptance formulas and decides which statements can be definitely accepted or rejected. The reduct implements the intuition that whatever is false in a stable model can be assumed false, but whatever is true in a stable model must be constructively provable. The next definition combines all of this.

Definition 2. Let $\Xi = (S, L, C)$ be an abstract dialectical framework. Define an operator by $\Gamma_\Xi(A, R) = (acc(A, R), rej(A, R))$ for $A, R \subseteq S$, where

$$acc(A, R) = \{s \in S \mid \text{for all } A \subseteq Z \subseteq (S \setminus R), \text{we have } Z \models \varphi_s\}$$
$$rej(A, R) = \{s \in S \mid \text{for all } A \subseteq Z \subseteq (S \setminus R), \text{we have } Z \not\models \varphi_s\}$$

For a set $M \subseteq S$, define the reduced ADF $\Xi^M = (M, L^M, C^M)$ by the set of links $L^M = L \cap (M \times M)$ and for each $s \in M$ we set $\varphi_s^M = \varphi_s[r/\bot : r \notin M]$. A model M for Ξ is a *stable model* of Ξ iff the least fixpoint of the operator Γ_{Ξ^M} is given by $(M, \emptyset)$.

Brewka and Woltran [4] showed that for any ADF Ξ, the operator Γ_Ξ always has a least fixpoint with respect to the component-wise $\subseteq$-ordering.[1] The computation of this least fixpoint starts with $(\emptyset, \emptyset)$; if it ends in $(M, \emptyset)$ then this intuitively means that all statements in M can be constructively derived assuming that all statements not in M are false. If the computation ends in anything other than $(M, \emptyset)$, then M is not a stable model.

Example 1 (Continued). Of the two models M_1, M_2 seen earlier, only M_2 is a stable model. Intuitively, the statement $b \in M_1$ cyclically supports itself.

It is clear that ADFs are a generalisation of AFs: for an argumentation framework $\Theta = (A, R)$, its *associated abstract dialectical framework* is given by $\Xi(\Theta) = (A, R, C)$ with $C_a(B) = in$ iff $B = \emptyset$ for each $a \in A$. But this is not just syntactical; Brewka and Woltran [4] showed that their semantical notions for

[1] That is, $(A, B) \leq (C, D)$ iff $A \subseteq C$ and $B \subseteq D$.

ADFs are generalisations of Dung's respective AF notions; likewise, in [5, 19] we proved correspondence results for all semantics defined there. Brewka and Woltran [4] defined a particular subclass of ADFs called *bipolar*. Intuitively, in bipolar ADFs each link is supporting or attacking (or both). It will turn out that ADFs resulting from our automatic translation from defeasible theory bases are all bipolar. This is especially significant as recent complexity results show that bipolar ADFs are as complex as AFs, thus the additional modelling capacities of bipolar ADFs come essentially for free [21].

3. Instantiations to Abstract Argumentation Frameworks

The general approach to provide a semantics for defeasible theories is to translate the defeasible theory into an argumentation formalism and then let the already existing semantics for that argumentation formalism determine the semantics of the defeasible theory. In the literature, the target formalism of choice are Dung's abstract argumentation frameworks. They abstract away from everything except arguments and attacks between them, so to define a translation to AFs one has to define arguments and attacks. We now review two particular such approaches.

3.1. The Approach of Caminada and Amgoud [6]

Caminada and Amgoud [6] define a translation from defeasible theories to argumentation frameworks. They create arguments in an inductive way by applying one or more inference rules. The internal structure of the arguments reflects how a particular conclusion was derived by applying an inference rule to the conclusions of subarguments, and allows arguments to be nested. So the base case of the induction takes into account rules with empty body, that is, rules of the form $\rightarrow \psi$ (or $\Rightarrow \psi$) for some literal ψ. Each such rule leads to an argument $A = [\rightarrow \psi]$ (or $[\Rightarrow \psi]$), and the conclusion of the rule becomes the conclusion of the argument. For the induction step, we assume there are arguments $A_1, \ldots, A_n$ with conclusions $\phi_1, \ldots, \phi_n$, respectively. If there is a strict rule $\phi_1, \ldots, \phi_n \rightarrow \psi$, we can build a new argument $A = [A_1, \ldots, A_n \rightarrow \psi]$ with conclusion ψ. (Likewise, if there is a defeasible rule $\phi_1, \ldots, \phi_n \Rightarrow \psi$, we can build a new argument $A = [A_1, \ldots, A_n \Rightarrow \psi]$.) Similar to rules, arguments can be strict or defeasible, where application of at least one defeasible rule makes the whole argument defeasible. In other words, strict arguments only use strict rules to derive their conclusion.

For these arguments, Caminada and Amgoud [6] then define two different kinds of attacks, rebuts and undercuts. An argument a rebuts another argument b if a subargument of a concludes some literal ψ, while there is a defeasible subargument of b that concludes $\overline{\psi}$. An argument a undercuts another argument b if the latter has a subargument that results from applying a defeasible rule and the applicability of that rule is disputed by a subargument of a. (So as a matter of principle, only defeasible arguments can be attacked.) Caminada and Amgoud [6] observed some difficulties of this translation.

Example 2 (Married John, [6, Example 4]). Consider the following vocabulary with intended natural-language meaning: $w \ldots$ John wears something that looks like a wedding ring, $g \ldots$ John often goes out late with his friends, $m \ldots$ John is married, $b \ldots$ John is a bachelor, $h \ldots$ John has a spouse. There are several relationships between these propositions, which are captured in the following theory base: the literals are $\mathit{Lit} = \{w, g, h, m, b, \neg w, \neg g, \neg h, \neg m, \neg b\}$, the strict rules are given by $\mathit{StrInf} = \{r_1 :\rightarrow w,\ r_2 :\rightarrow g,\ r_3 : b \rightarrow \neg h,\ r_4 : m \rightarrow h\}$ and the defeasible rules $\mathit{DefInf} = \{r_5 : w \Rightarrow m,\ r_6 : g \Rightarrow b\}$.

In the ASPIC system of Caminada and Amgoud [6], all the literals in the set $S = \{w, g, m, b\}$ are contained in all extensions (with respect to any of Dung's standard semantics) of the constructed AF. Caminada and Amgoud observe that this is clearly unintended since the natural-language interpretation would be that John is a married bachelor. Moreover, the closure of S under StrInf is $\mathit{Cl}_{\mathit{StrInf}}(S) = \{w, g, m, b, h, \neg h\}$, which is inconsistent. So not only are there applicable strict rules that have not been applied in S, but their application would lead to inconsistency.

To avoid anomalies such as the one just seen, Caminada and Amgoud [6] went on to define three natural rationality postulates for rule-based argumentation-based systems that are concerned with the interplay of consistency and strict rule application. Our formulation of them is slightly different for various reasons:

- We are concerned with argumentation frameworks as well as with abstract dialectical frameworks in this paper, so we made the postulates parametric in the target argumentation formalism.
- We removed the respective second condition on the sceptical conclusions with respect to all extensions/models. Propositions 4 and 5 in [6] show that they are redundant in their case.
- We are not constrained to formalisms and semantics where there are only finitely many extensions/models.
- For the sake of readability, we assume that the literals Lit of the defeasible theory are contained in the vocabulary of the target formalism.[2]

The first postulate requires that the set of conclusions for any extension should be closed under application of strict rules.

Postulate 1 (Closure). Let $(\mathit{Lit}, \mathit{StrInf}, \mathit{DefInf})$ be a defeasible theory. Its translation satisfies *closure* for semantics σ iff for any σ-model M, we find that $\mathit{Cl}_{\mathit{StrInf}}(\mathit{Lit} \cap M) \subseteq \mathit{Lit} \cap M$.

[2] This is not a proper restriction since reconstruction of conclusions about the original defeasible theory is one of the goals of the whole enterprise and so there should be at least a translation function from argumentation models to theory models.

Naturally, the notion of consistency is reduced to consistency of a set of literals of the underlying logical language. Note that consistency only concerns the local consistency of a given single model of the target formalism. It may well be that the formalism is globally inconsistent in the sense of not allowing for any model with respect to a particular semantics. The latter behaviour can be desired, for example if the original theory base is inconsistent already.

Postulate 2 (Direct Consistency). Let $(Lit, StrInf, DefInf)$ be a defeasible theory with translation X and σ a semantics. X satisfies *direct consistency* iff for all σ-models M we have that $Lit \cap M$ is consistent.

Caminada and Amgoud [6] remark that it is usually easy to satisfy direct consistency, but much harder to satisfy the stronger notion of indirect consistency. For this to hold, for each model its closure under strict rules must be consistent.

Postulate 3 (Indirect Consistency). Let $(Lit, StrInf, DefInf)$ be a defeasible theory with translation X and σ a semantics. X satisfies *indirect consistency* iff for all σ-models M we have that $Cl_{StrInf}(Lit \cap M)$ is consistent.

As a counterpart to Proposition 7 of Caminada and Amgoud [6], we can show that closure and direct consistency together imply indirect consistency.

Proposition 1. *Let $(Lit, StrInf, DefInf)$ be a defeasible theory with translation X and σ a semantics. If X satisfies closure and direct consistency, then it satisfies indirect consistency.*

Proof. Let X satisfy closure and direct consistency, and let M be a σ-model for X. We have to show that $Cl_{StrInf}(Lit \cap M)$ is consistent. Since X satisfies closure, $Cl_{StrInf}(Lit \cap M) \subseteq Lit \cap M$. Now since X satisfies direct consistency, $Lit \cap M$ is consistent. Hence its subset $Cl_{StrInf}(Lit \cap M) \subseteq Lit$ is consistent and X satisfies indirect consistency. □

While Caminada and Amgoud [6] observed problematic issues in giving argument-based semantics to defeasible theory bases, they still succeeded in devising an approach that is able to achieve closure and direct and indirect consistency for any admissibility-based semantics by using appropriate definitions of rebut and undercut.

3.2. The Approach of Wyner et al. [24]

Wyner et al. [23, 24] identified some problems of the approach of Caminada and Amgoud [6] and proposed an alternative translation from theory bases to argumentation frameworks. We do not necessarily support or reject their philosophical criticisms, but rather find the translation technically appealing. They create an argument for each literal in the theory base's language and additionally an argument for each rule. Intuitively, the literal arguments indicate that the literal holds, and the rule arguments indicate that the rule is applicable. Furthermore, the defined conflicts between these arguments are straightforward:

(1) opposite literals attack each other; (2) rules are attacked by the negations of their body literals; (3) defeasible rules are attacked by the negation of their head; (4) all rules attack the negation of their head.

Definition 3 (Definitions 4 and 5 in [23]). Let $TB = (Lit, StrInf, DefInf)$ be a defeasible theory. Define an argumentation framework $\Theta(TB) = (A, R)$ by

$$\begin{aligned} A = \quad & Lit \cup \{r \mid r : \phi_1, \ldots, \phi_n \Rrightarrow \psi \in StrInf \cup DefInf\} \\ R = \quad & \{(\psi, \overline{\psi}) \mid \psi \in Lit\} \\ \cup \; & \{(\overline{\phi_i}, r) \mid r : \phi_1, \ldots, \phi_n \Rrightarrow \psi \in StrInf \cup DefInf, 1 \leq i \leq n\} \\ \cup \; & \{(\overline{\psi}, r) \mid r : \phi_1, \ldots, \phi_n \Rightarrow \psi \in DefInf\} \\ \cup \; & \{(r, \overline{\psi}) \mid r : \phi_1, \ldots, \phi_n \Rrightarrow \psi \in StrInf \cup DefInf\} \end{aligned}$$

As mentioned in the introduction, a further definition is needed to rule out extensions that are not closed under strict rules.

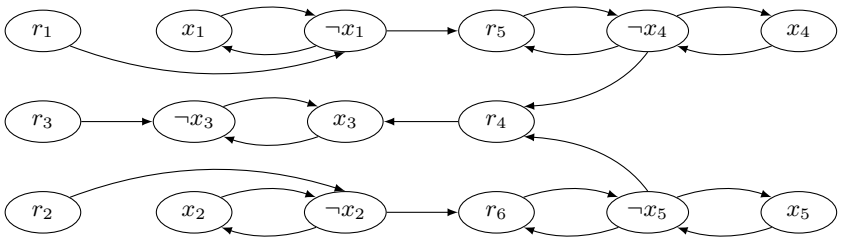
Definition 4 (Definition 7 in [24]). Let $TB = (Lit, StrInf, DefInf)$ be a defeasible theory and $\Theta(TB) = (A, R)$ its associated argumentation framework. An extension $M \subseteq A$ of $\Theta(TB)$ is *well-formed* if there is no strict rule $r : \phi_1, \ldots, \phi_n \to \psi \in StrInf$ such that $\{r, \phi_1, \ldots, \phi_n\} \subseteq M$ but $\psi \notin M$.

It is decidable in polynomial time whether a given extension M is well-formed: we can compute $Cl_{StrInf}(M)$ and then check whether $Cl_{StrInf}(M) \subseteq M$. This means that the additional computational cost incurred by Definition 4 is acceptable under standard assumptions.[3] For illustration, let us now look at one of the examples of Wyner et al. [24] which they adapted from [6].

Example 3 (Example 4 in [24]). Consider the following theory base.

$$\begin{aligned} Lit &= \{x_1, x_2, x_3, x_4, x_5, \neg x_1, \neg x_2, \neg x_3, \neg x_4, \neg x_5\} \\ StrInf &= \{r_1 :\to x_1, \quad r_2 :\to x_2, \quad r_3 :\to x_3, \quad r_4 : x_4, x_5 \to \neg x_3\} \\ DefInf &= \{r_5 : x_1 \Rightarrow x_4, \quad r_6 : x_2 \Rightarrow x_5\} \end{aligned}$$

We can see that x_1, x_2, x_3 are strictly asserted and thus should be contained in any extension. The AF translation is depicted below.



[3] It might even be possible to encode Definition 4 directly into the translated AF by adding, for each strict rule $r : \phi_1, \ldots, \phi_n \to \psi \in StrInf$, a new argument -$r$ ("r is inapplicable") and the attacks (-r, -r), (ψ, -r), ($\overline{\phi_1}$, -r), ..., ($\overline{\phi_n}$, -r).

The stable extensions of this AF are as follows:

$$S_1 = \{x_1, x_2, x_3, \neg x_4, \neg x_5, r_1, r_2, r_3\} \quad S_2 = \{x_1, x_2, x_3, \neg x_4, x_5, r_1, r_2, r_3, r_6\}$$
$$S_3 = \{x_1, x_2, x_3, x_4, \neg x_5, r_1, r_2, r_3, r_5\} \quad S_4 = \{x_1, x_2, x_4, x_5, r_1, r_2, r_3, r_4, r_5, r_6\}$$

While the first three extensions can be considered intended, S_4 is not closed under strict rules and indirectly inconsistent: r_3 is applicable but x_3 does not hold, r_4 is applicable but $\neg x_3$ does not hold. Indeed, S_4 is not well-formed and thus should not be considered for drawing conclusions [24].

A similar observation can be made in Example 2: the AF translation according to Wyner et al. [23] has a stable extension $\{w, g, m, b, r_1, r_2, r_3, r_4, r_5, r_6\}$ where John is a married bachelor; but again, this extension is not well-formed and thus discarded.

4. Instantiations to Abstract Dialectical Frameworks

In this section, we extend the theory base to AF translation of Wyner et al. [23] to ADFs. Due to the availability of support, this is straightforward. Indeed, support and attack are sufficient for our purposes and we can therefore restrict our attention to bipolar ADFs.

4.1. From Theory Bases to ADFs

As in the approach of Wyner et al. [23], we directly use the literals from the theory base as statements that express whether the literal holds. We also use rule names as statements indicating that the rule is applicable. Additionally, for each rule r we create a statement -r indicating that the rule has not been applied. Not applying a rule is acceptable for defeasible rules, but unacceptable for strict rules since it would violate the closure postulate. This is enforced via integrity constraints saying that it may not be the case in any model that the rule body holds but the head does not hold: Technically, for a strict rule r, we introduce a conditional self-attack of -r; this self-attack becomes active if (and only if) the body of r is satisfied but the head of r is not satisfied, thereby preventing this undesirable state of affairs from getting included in a model. Defeasible rules offer some degree of choice, whence we leave it to the semantics whether or not to apply them. This choice is modelled by a mutual attack cycle between r and -r. The remaining acceptance conditions are equally straightforward:

- Opposite literals attack each other.
- A literal is accepted whenever some rule deriving it is applicable, that is, all rules with head ψ support statement ψ.
- A strict rule is applicable whenever all of its body literals hold, that is, the body literals of r are exactly the supporters of r.

- Likewise, a defeasible rule is applicable whenever all of its body literals hold, and additionally the negation of its head literal must not hold.

In particular, literals cannot be accepted unless there is some rule deriving them.

Definition 5. Let $TB = (Lit, StrInf, DefInf)$ be a theory base. Define an ADF $\Xi(TB) = (S, L, C)$ by $S = Lit \cup \{r, \text{-}r \mid r : \phi_1, \ldots, \phi_n \Rrightarrow \psi \in StrInf \cup DefInf\}$; the acceptance functions of statements s can be parsimoniously represented by propositional formulas φ_s.[4] For a literal $\psi \in Lit$, we define

$$\varphi_\psi = \neg[\overline{\psi}] \wedge \bigvee_{r:\phi_1,\ldots,\phi_n \Rrightarrow \psi \in StrInf \cup DefInf} [r]$$

For a strict rule $r : \phi_1, \ldots, \phi_n \to \psi \in StrInf$, we define

$$\varphi_r = [\phi_1] \wedge \ldots \wedge [\phi_n] \qquad \text{and} \qquad \varphi_{\text{-}r} = [\phi_1] \wedge \ldots \wedge [\phi_n] \wedge \neg[\psi] \wedge \neg[\text{-}r]$$

For a defeasible rule $r : \phi_1, \ldots, \phi_n \Rightarrow \psi \in DefInf$, we define

$$\varphi_r = [\phi_1] \wedge \ldots \wedge [\phi_n] \wedge \neg[\overline{\psi}] \wedge \neg[\text{-}r] \qquad \text{and} \qquad \varphi_{\text{-}r} = \neg[r]$$

Finally, there is a link $(s', s) \in L$ iff $[s']$ occurs in the acceptance formula φ_s.

(For the formulas defined above, the empty disjunction leads to $\bot$ – logical falsity – and the empty conjunction to $\top$ – logical truth.)

Let us see how our translation treats the examples seen earlier.

Example 3 (Continued). Definition 5 yields these acceptance formulas:

$$\begin{array}{llll}
\varphi_{x_1} = \neg[\neg x_1] \wedge [r_1] & \varphi_{x_2} = \neg[\neg x_2] \wedge [r_2] & \varphi_{x_3} = \neg[\neg x_3] \wedge [r_3] & \\
\varphi_{x_4} = \neg[\neg x_4] \wedge [r_5] & \varphi_{x_5} = \neg[\neg x_5] \wedge [r_6] & & \\
\varphi_{\neg x_1} = \bot \quad \varphi_{\neg x_2} = \bot & \varphi_{\neg x_3} = \neg[x_3] \wedge [r_4] & \varphi_{\neg x_4} = \bot & \varphi_{\neg x_5} = \bot \\
\varphi_{r_1} = \top \quad \varphi_{r_2} = \top & \varphi_{r_3} = \top & \varphi_{r_4} = [x_4] \wedge [x_5] & \\
\varphi_{r_5} = [x_1] \wedge \neg[\neg x_4] \wedge \neg[\text{-}r_5] & \varphi_{r_6} = [x_2] \wedge \neg[\neg x_5] \wedge \neg[\text{-}r_6] & & \\
\varphi_{\text{-}r_1} = \neg[x_1] \wedge \neg[\text{-}r_1] & \varphi_{\text{-}r_2} = \neg[x_2] \wedge \neg[\text{-}r_2] & \varphi_{\text{-}r_3} = \neg[x_3] \wedge \neg[\text{-}r_3] & \\
\varphi_{\text{-}r_4} = [x_4] \wedge [x_5] \wedge \neg[\neg x_3] \wedge \neg[\text{-}r_4] & \varphi_{\text{-}r_5} = \neg[r_5] & \varphi_{\text{-}r_6} = \neg[r_6] &
\end{array}$$

Statements with an acceptance condition of the form $\neg p_1 \wedge \ldots \wedge \neg p_n$ behave like AF arguments. So in particular r_1, r_2, r_3 are always *in* since these rules have an empty body. Similarly, $\text{-}r_1, \text{-}r_2, \text{-}r_3$ are self-attacking arguments. The statements $\neg x_1, \neg x_2, \neg x_4, \neg x_5$ are always *out* since there are no rules deriving these literals. The remaining acceptance conditions are clear from the definitions: literals are

[4] In these formulas, we write ADF statements in brackets, to avoid confusion between negation being applied inside a statement name – as in $[\neg x]$ – and negation being applied in the formula outside of the statement's name – as in $\neg[\text{-}r]$. Thus $[\neg x]$ and $\neg[x]$ are syntactically different literals in the language of acceptance formulas; their meaning is intertwined via the semantics of ADFs.

supported by the rules deriving them and rules in turn are supported by their body literals.

For this ADF, models and stable models coincide, and there are three of them:

$$M_1 = \{x_1, x_2, x_3, r_1, r_2, r_3, \text{-}r_5, \text{-}r_6\} \qquad M_2 = \{x_1, x_2, x_3, x_4, r_1, r_2, r_3, r_5, \text{-}r_6\}$$
$$M_3 = \{x_1, x_2, x_3, x_5, r_1, r_2, r_3, \text{-}r_5, r_6\}$$

Roughly, in M_1 none of the defeasible rules r_5, r_6 has been applied – indicated by $\text{-}r_5$ and $\text{-}r_6$ –, while in M_2 and M_3 either one of them has been applied. As intended, there is no model where both defeasible rules have been applied, as this would lead to a set that contains both x_4 and x_5; this in turn would make rule r_4 applicable, allowing to conclude $\neg x_3$ in contradiction to x_3 being strictly true according to rule r_3. We can furthermore see that all of the models are closed under strict rule application (they contain x_1, x_2, x_3 and no other strict rule is applicable) and directly consistent, thus also indirectly consistent.

A similar observation can be made for John (not) being married (Example 2); our ADF translation has three (stable) models: $M_1 = \{w, g, r_1, r_2, \text{-}r_5, \text{-}r_6\}$, $M_2 = \{w, g, h, m, r_1, r_2, r_4, r_5, \text{-}r_6\}$ and $M_3 = \{w, g, b, \neg h, r_1, r_2, r_3, \text{-}r_5, r_6\}$. Again, the argumentation translation of the theory base satisfies closure and direct and indirect consistency. We will later prove that the satisfaction of the postulates is not a coincidence in our approach. But first of all let us consider another problem which often arises in knowledge representation and reasoning.

4.2. Support Cycles in Theory Bases

When logical, rule-based approaches are used for knowledge representation, a recurring issue is that of cyclic dependencies between propositions of the knowledge base. If such support cycles are carelessly overlooked or otherwise not treated in an adequate way, they can lead to counterintuitive conclusions. Consider this famous example by Denecker et al. [8].

Example 4 (Gear Wheels [8]). There are two interlocked gear wheels x and y that can be separately turned and stopped. Let x_0 and y_0 denote whether x (resp. y) turns at time point 0, and likewise for a successive time point 1. At any one time point, whenever the first wheel turns (resp. stops), it causes the second one to turn (resp. stop), and vice versa. This is expressed by strict rules r_1 to r_8. Without a cause for change, things usually stay the way they are from one time point to the next, which is expressed by the defeasible rules r_a to r_d.

$$\begin{aligned}
\mathit{Lit} &= \{x_0, y_0, x_1, y_1, \neg x_0, \neg y_0, \neg x_1, \neg y_1\} \\
\mathit{StrInf} &= \{r_1 : x_0 \to y_0, \quad r_2 : y_0 \to x_0, \quad r_3 : \neg x_0 \to \neg y_0, \quad r_4 : \neg y_0 \to \neg x_0, \\
&\qquad r_5 : x_1 \to y_1, \quad r_6 : y_1 \to x_1, \quad r_7 : \neg x_1 \to \neg y_1, \quad r_8 : \neg y_1 \to \neg x_1\} \\
\mathit{DefInf} &= \{r_a : x_0 \Rightarrow x_1, \quad r_b : \neg x_0 \Rightarrow \neg x_1, \quad r_c : y_0 \Rightarrow y_1, \quad r_d : \neg y_0 \Rightarrow \neg y_1\}
\end{aligned}$$

For later reference, we denote this theory base by $TB_{GW} = (\mathit{Lit}, \mathit{StrInf}, \mathit{DefInf})$. To model a concrete scenario, we add the rules

$StrInf' = \{r_i :\rightarrow \neg x_0, r_j :\rightarrow \neg y_0\}$ expressing that both wheels initially stand still. We denote the augmented theory base for this concrete scenario by $TB'_{GW} = (Lit, StrInf \cup StrInf', DefInf)$. It is clearly unintended that there is some model for TB'_{GW} where the gear wheels magically start turning with one being the cause for the other and vice versa.

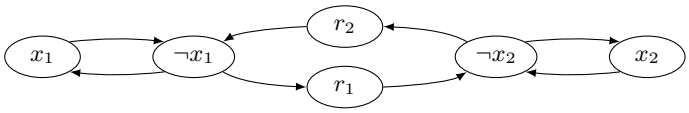
Example 5 (Defeasible cycle). Consider these defeasible rules saying that *rain* and *wet* grass usually go hand in hand: $Lit = \{rain, wet, \neg rain, \neg wet\}$, $StrInf = \emptyset$ and $DefInf = \{r_1 : rain \Rightarrow wet, r_2 : wet \Rightarrow rain\}$. The intended meaning is that one is usually accompanied by the other, not that both may appear out of thin air.

To see how argumentation translations of theory bases treat such cycles, let us look at a simplified version of the gear wheels example.

Example 6 (Strict cycle). Consider a theory base with two literals mutually supporting each other through strict rules: $Lit = \{x_1, x_2, \neg x_1, \neg x_2\}$, the strict rules are given by $StrInf = \{r_1 : x_1 \rightarrow x_2, \quad r_2 : x_2 \rightarrow x_1\}$ and $DefInf = \emptyset$. Our ADF translation of this example yields the acceptance formulas

$$\varphi_{x_1} = [r_2] \qquad \varphi_{\neg x_1} = \bot \qquad \varphi_{r_1} = [x_1] \qquad \varphi_{\text{-}r_1} = [x_1] \wedge \neg[x_2] \wedge \neg[\text{-}r_1]$$
$$\varphi_{x_2} = [r_1] \qquad \varphi_{\neg x_2} = \bot \qquad \varphi_{r_2} = [x_2] \qquad \varphi_{\text{-}r_2} = [x_2] \wedge \neg[x_1] \wedge \neg[\text{-}r_2]$$

The ADF has two models, $M_1 = \{x_1, x_2, r_1, r_2\}$ and $M_2 = \emptyset$. Only M_2 is a stable model due to the cyclic self-support of the statements in M_1. Note that not only do x_1 and x_2 not hold in M_2, neither do $\neg x_1$ and $\neg x_2$ (there are no rules possibly deriving them). In contrast, the translation of Wyner et al. [23] yields the AF



with two stable extensions $S_1 = \{x_1, r_1, x_2, r_2\}$ and $S_2 = \{\neg x_1, \neg x_2\}$. In S_1, x_1 and x_2 hold due to self-support while in S_2 they are "guessed" to be false.[5]

In our view, this is problematic since it is not made clear to the user that these different extensions arise due to self-support. Even if we grant that for some application domains, cyclic self-support of literals might be intended or at least not unintended, the user should be able to distinguish whether different models/extensions arise due to present or absent self-support on the one hand, or due to conflicts between defeasible conclusions on the other hand. ADFs provide this important distinction, since cycles are allowed in models and disallowed in

[5] In this paper, we consider only stable extension semantics for AFs. It might be possible to choose/come up with an AF semantics that treats the above AF differently.

stable models, while both semantics are identical in their treatment of conflicts between defeasible conclusions.

In the approach of Caminada and Amgoud [6], treatment of cycles is built into the definition of the set of arguments in the resulting argumentation framework. The arguments are created using structural induction, where rules with empty bodies form the induction base and all other rules form the induction step. For the general gear wheel domain TB_{GW} of Example 4, and for Examples 5 and 6, their translation would not create any arguments (there are no assertions in the theory bases), and the approach could not draw any conclusions about these examples. The concrete scenario of the interlocked gear wheel domain TB'_{GW} in Example 4, where both wheels initially stand still, would be treated correctly by the approach of Caminada and Amgoud [6]. But note that the well-foundedness of the treatment of cyclic dependencies is built into the syntax of the resulting argumentation framework – there are no arguments that could conclude that any of the wheels is turning, although there are (strict and defeasible) rules with such conclusions.[6] Consequently, a part of the semantics of the theory base is already fixed by the translation, irrespective of the argumentation semantics that is used later on.

4.3. Inconsistent Theory Bases

Example 7 (Inconsistent Theory Base). Consider the following (obviously inconsistent) theory base in which both a literal and its negation are strictly asserted: $\mathit{Lit} = \{x, \neg x\}$, $\mathit{StrInf} = \{r_1 :\rightarrow x,\ r_2 :\rightarrow \neg x\}$ and $\mathit{DefInf} = \emptyset$. Our ADF translation yields the acceptance formulas

$$\varphi_x = \neg[\neg x] \wedge [r_1] \qquad \varphi_{r_1} = \top \qquad \varphi_{\text{-}r_1} = \neg[x] \wedge \neg[\text{-}r_1]$$
$$\varphi_{\neg x} = \neg[x] \wedge [r_2] \qquad \varphi_{r_2} = \top \qquad \varphi_{\text{-}r_2} = \neg[\neg x] \wedge \neg[\text{-}r_2]$$

This ADF has no models, and so the theory base's inconsistency is detected.

On the other hand, the associated argumentation framework due to Wyner et al. [23] is given by the set of arguments $A = \{x, \neg x, r_1, r_2\}$ and the attacks $R = \{(x, \neg x), (\neg x, x), (r_1, \neg x), (r_2, x)\}$. In the only stable extension $\{r_1, r_2\}$ both rules are applicable but none of the head literals hold due to immanent conflict. Again, this extension is not well-formed and the inconsistency is made obvious.

In the approach of Caminada and Amgoud [6], we can construct two strict arguments that conclude x and $\neg x$, respectively. There are no attacks between these arguments, since rebuts are impossible between strict arguments and rules without body cannot be undercut. So their resulting AF has a stable extension from which both x and $\neg x$ can be concluded, which detects the inconsistency.

[6] See also the discussion of (non-)treatment of *partial* knowledge bases by Wyner et al. [24].

4.4. Properties of the Translation

In this section, we analyse some theoretical properties of our translation. First we show that it satisfies (our reformulations of) the rationality postulates of Caminada and Amgoud [6]. Then we analyse the computational complexity of translating a given theory base and show that the blowup is at most quadratic.

Postulates. It is elementary to show that the ADFs resulting from our translation satisfy direct consistency. This is because the statements ψ and $\overline{\psi}$ mutually attack each other.

Proposition 2. *For any theory base $TB = (Lit, StrInf, DefInf)$, its associated ADF $\Xi(TB)$ satisfies direct consistency with respect to the model semantics.*

Proof. Let M be a model for $\Xi(TB)$ and assume to the contrary that $M \cap Lit$ is inconsistent. Then there is a $\psi \in Lit$ such that $\psi \in M$ and $\neg\psi \in M$. Since $\neg\psi \in M$, the acceptance condition of $\neg\psi$ yields $\psi \notin M$. Contradiction. □

We can also prove that they satisfy closure: by construction, the (acceptance conditions of) statements $\text{-}r$ for strict rules r guarantee that the rule head is contained in any model that contains the rule body.

Proposition 3. *For any theory base $TB = (Lit, StrInf, DefInf)$, its associated ADF $\Xi(TB)$ satisfies closure with respect to the model semantics.*

Proof. Let M be a model of $\Xi(TB)$ and $r : \phi_1, \ldots, \phi_n \to \psi \in StrInf$ such that we find $\phi_1, \ldots, \phi_n \in M$. We have to show $\psi \in M$. By definition, $\Xi(TB)$ has a statement $\text{-}r$ with parents $par(\text{-}r) = \{\phi_1, \ldots, \phi_n, \psi, \text{-}r\}$. We next show that $\text{-}r \notin M$: assume to the contrary that $\text{-}r \in M$. Then by the acceptance condition of $\text{-}r$ we get $\text{-}r \notin M$, contradiction. Thus $\text{-}r \notin M$. Now the acceptance condition of $\text{-}r$ yields $\phi_1 \notin M$ or $\ldots$ or $\phi_n \notin M$ or $\psi \in M$ or $\text{-}r \in M$. By assumption, we have $\phi_1, \ldots, \phi_n \in M$ and $\text{-}r \notin M$, thus we get $\psi \in M$. □

By Proposition 1 the translation satisfies indirect consistency.

Corollary 4. *For any theory base $TB = (Lit, StrInf, DefInf)$, its associated ADF $\Xi(TB)$ satisfies indirect consistency with respect to the model semantics.*

Since any stable model is a model, our translation also satisfies the postulates for the stable model semantics.

Corollary 5. *For any theory base $TB = (Lit, StrInf, DefInf)$, its associated ADF $\Xi(TB)$ satisfies closure and direct and indirect consistency with respect to the stable model semantics.*

It should be noted that defeasible rules may or may not be applied – the approach is not eager to apply defeasible rules.

Complexity. For a theory base $TB = (Lit, StrInf, DefInf)$, we define the size of its constituents as follows. Quite straightforwardly, the size of a set of literals is just its cardinality, the size of a rule is the number of literals in it, the size of a set of rules is the sum of the sizes of its elements and the size of a theory base is the sum of the sizes of its components.

We want to analyse the size of its ADF translation $\Xi(TB) = (S, L, C)$ according to Definition 5. Clearly, the number of statements is linear in the size of the theory base, since we have one statement for each literal and two statements for each rule: $|S| = |Lit| + 2 \cdot (|StrInf| + |DefInf|)$. Since $L \subseteq S \times S$, the number of links in L is at most quadratic in the cardinality of S: $|L| \leq |S|^2$. Finally, we have seen in Definition 5 that the acceptance conditions of statements can be parsimoniously represented by propositional formulas. It can be checked that the size of each one of these formulas is at most linear in the size of the theory base. Since there are linearly many statements with one acceptance formula each, the acceptance conditions can be represented in quadratic space. So overall, the resulting ADF $\Xi(TB) = (S, L, C)$ can be represented in space which is at most quadratic in the size of the original theory base. In particular, in our approach a finite theory base always yields a finite argumentation translation. This is in contrast to the definition of Caminada and Amgoud [6], where the strict rule set $StrInf = \{r_0 :\rightarrow a, r_1 : a \rightarrow b, r_2 : b \rightarrow a\}$ allows to construct infinitely many arguments $A_1 = [\rightarrow a], A_2 = [A_1 \rightarrow b], A_3 = [A_2 \rightarrow a], A_4 = [A_3 \rightarrow b], \ldots$[7]

5. A Direct Semantics for Defeasible Theory Bases

We have seen previously how ADFs can be used to give a semantics to defeasible theory bases. Albeit we introduced additional, merely technical statements (like -r), we were able to address shortcomings of previous approaches. Still, there remains the issue that the ADF-based semantics is not necessarily eager to apply defeasible rules. In what follows, we will introduce a direct semantics for defeasible theory bases that possesses this eagerness property. It will additionally allow us to more precisely clarify our intuitions about what rules mean, especially the difference between strict and defeasible rules. While our intuitions on defeasible rules are quite clear, we will argue that there are two different intuitions on strict rules. One intuition says that strict rules are directed inference rules that operate on the knowledge level, that is, whenever the premises are known then the conclusion is inferred. In particular, in being directed these rules do not automatically entail any of their contrapositives. Let us call this intuition (DR) for *directed inference rule*; we will see that (DR) can lead to problems with global inconsistency. Another intuition says that strict rules are just like material implications in propositional logic, let us call it (MI). In par-

[7] Even if we exclude cycles in rules, there are rule sets that allow for exponentially many arguments: Set $D_0 = \{\Rightarrow p_0, \Rightarrow \neg p_0\}$, $D_1 = D_0 \cup \{p_0 \Rightarrow p_1, \neg p_0 \Rightarrow p_1\}$ and for $i \geq 1$, $D_{i+1} = D_i \cup \{p_0, p_i \Rightarrow p_{i+1}, \neg p_0, p_i \Rightarrow p_{i+1}\}$. For any $n \in \mathbb{N}$, the size of D_n is linear in n and D_n leads to 2^{n+1} arguments, among them 2^n arguments for p_n.

ticular, in this intuition strict rules are not directed and therefore equivalent to their contrapositives.[8] (MI) is unproblematic in its interaction with defeasible rules, but raises the philosophical question why strict rules should allow for contraposition and defeasible rules should not. These questions are pervasive in giving semantics to nonmonotonic rule-based systems, and may account for (parts of) the complications encountered by Caminada and Amgoud [6].

To formalise the two mentioned intuitions, we make use of concepts from epistemic modal logic. We consider epistemic states in the form of sets of possible worlds, where a possible world is simply a two-valued interpretation of a propositional vocabulary. More precisely, let A be a propositional signature. Then an interpretation over A can be represented as a set $w \subseteq A$ as usual; we will also call an interpretation a *world*. We then define the *set of worlds over* A as $W_A \stackrel{\text{def}}{=} 2^A$. A set $Q \subseteq W_A$ is then an *epistemic state*: intuitively, any entity being in the epistemic state Q considers exactly the worlds $w \in Q$ to be possible, that is, to be the one single world the entity "lives in." Put another way, an epistemic state Q signifies that any entity subscribing to this epistemic state cannot distinguish the worlds in Q with what it knows. (But it can distinguish worlds *in* Q from those *not in* Q.) The knowledge associated with an epistemic state Q over A is simply the set of propositional formulas over A which are true in all possible worlds, the theory $\{\varphi \mid w \models \varphi \text{ for all } w \in Q\}$.

We start to formalise the intuition (MI), where strict rules $\phi_1, \ldots, \phi_n \to \psi$ are interpreted as material implications $(\phi_1 \wedge \ldots \wedge \phi_n) \supset \psi$ in propositional logic. To do this, we define a satisfaction relation $\models$, that indicates whether an epistemic state together with a specific world (the "real world") satisfies an element of a defeasible theory base. Of course, it is trivial to define this for literals. For strict rules, the real world must satisfy the above material implication. For defeasible rules $r : \phi_1, \ldots, \phi_n \Rightarrow \psi$, our intuition is as follows: Assume that w is the real world and Q is our epistemic state. If we *know* that all body literals $\phi_1, \ldots, \phi_n$ hold, and we do *not* know that the conclusion is false, then for the pair Q, w to satisfy the defeasible rule, the conclusion must hold in the real world w. Otherwise, quite simply, the defeasible rule would not be a very valuable guide on what normally holds in the world. In our formalisation below, this intuition is split up into three ways how a defeasible rule can be satisfied:

1. Not all of the body literals are known. (Then the rule is inapplicable due to insufficient premises.)
2. The negation of the head literal is known. (Then the rule is inapplicable due to an exception.)
3. The head literal is actually true. (Then the defeasible rule is good regardless of what we know, because it tells us something true about the world.)

Definition 6. Let $TB = (Lit, StrInf, DefInf)$ be a defeasible theory. Let

[8] Caminada and Amgoud have a similar concept, *closure under transposition* [6, Def. 17].

$A \subseteq \mathit{Lit}$ be all atoms of the language, $a \in A$, $w \in W_A$ and $Q \subseteq W_A$.[9]

$w \models a$	iff $a \in w$
$w \models \neg a$	iff $a \notin w$
$w \models r : \phi_1, \ldots, \phi_n \to \psi$	iff $w \models (\phi_1 \wedge \ldots \wedge \phi_n) \supset \psi$ in propositional logic
$Q, w \models r : \phi_1, \ldots, \phi_n \Rightarrow \psi$	iff there is a $v \in Q$ and $1 \leq i \leq n$ with $v \not\models \phi_i$
	or for all $v \in Q$ we have $v \not\models \psi$
	or $w \models \psi$
$Q, w \models \mathit{TB}$	iff $Q, w \models r$ for all $r \in \mathit{StrInf} \cup \mathit{DefInf}$

We use Examples 1 and 2 from [24] to illustrate the definitions.

Example 8 (Partial theories). Consider the set of literals $\mathit{Lit} = \{x_1, x_2, \neg x_1, \neg x_2\}$; then the set of atoms is $A = \{x_1, x_2\}$. Consequently, there are four possible worlds, that is, $W_A = \{\emptyset, \{x_1\}, \{x_2\}, \{x_1, x_2\}\}$. It follows that 2^{W_A} contains $2^4 = 16$ different epistemic states, among them the state W_A where any world is considered possible (thus the agent knows nothing) and the state $\emptyset$ where the agent's knowledge is inconsistent.

Considering the strict rule $r_1 : x_1 \to x_2$, it is easy to see that it is satisfied by all worlds except $\{x_1\}$. For its defeasible variant $r_2 : x_1 \Rightarrow x_2$ we have the following: Assume the epistemic state $Q = \{\{x_1\}, \{x_1, x_2\}\}$ where we know that x_1 is true but are oblivious whether x_2 holds, and the real world $w = \{x_1\}$ where x_2 is false. Then we have $Q, w \not\models r_2 : x_1 \Rightarrow x_2$ since we know that the rule's body is true, do not know that its head is false, but its head is false in the real world. For $w' = \{x_1, x_2\}$, we would get $Q, w' \models r_2 : x_1 \Rightarrow x_2$ since $w' \models x_2$.

With the satisfaction relation at hand, it is then straightforward to define when an epistemic state Q is a model of a defeasible theory: whenever Q coincides with the set of possible worlds w for which the pair Q, w satisfies all rules in the defeasible theory base.

Definition 7. For a theory base TB, a set $Q \subseteq W_A$ of possible worlds is a *model for* TB if and only if $Q = \{w \in W_A \mid Q, w \models \mathit{TB}\}$.

Example 8 (Continued). For the defeasible theory base TB_1 consisting only of the strict rule $r_1 : x_1 \to x_2$, we get a single model $Q_1 = \{\emptyset, \{x_2\}, \{x_1, x_2\}\}$. In Q_1 we know that x_1 implies x_2, but we do not know anything else. These possible worlds correspond one-to-one with the preferred extensions that Wyner et al. obtain for the very same theory [24, Example 1].

For the defeasible theory base TB_2 consisting only of the defeasible rule $r_2 : x_1 \Rightarrow x_2$, the only model is $Q_2 = W_A$ where all worlds are considered possible. Intuitively, the premise of the defeasible rule is not known, and so the rule cannot be applied.

[9] For conciseness, we leave out the epistemic state or the real world when it is not used in the definition of the satisfaction relation.

Let us consider some further examples.

Example 2 (Continued). For the married John example, we get two models:

$$Q_1 = \{\{g, w, b\}\} \quad \text{and} \quad Q_2 = \{\{g, w, m, h\}\}$$

In both models, the epistemic state is fully determined, that is, we know exactly which world is the real one. In Q_1, John is a bachelor; in Q_2, he is married and thus has a spouse. In both epistemic states, John goes out and wears a ring. Note that the semantics is eager to apply defeasible rules – while Q_1 and Q_2 directly correspond to the models M_3 and M_2 (page 12) of the ADF translation, there is no possible-worlds equivalent of M_1 where no defeasible rule has been applied. The reason for this is easy to see: if the epistemic state $Q_3 = \{\{g, w\}\}$ were a model, then we would have $Q_3 = \{w \in W_A \mid Q_3, w \models TB\}$. However for $v' = \{g, w, b\}$, we find that $Q_3, v' \models TB$ but $v' \notin Q_3$. Intuitively, the pair Q_3, v' satisfies the defeasible rule $r_6 : g \Rightarrow b$ because $v' \models b$; so according to what is known v' should be considered a possible world, but Q_3 does not do so. (The same can be shown for $v'' = \{g, w, m, h\}$.)

The behaviour of the other problematic example follows suit.

Example 3 (Continued). Again, we get two models:

$$Q_1 = \{\{x_1, x_2, x_3, x_4\}\} \quad \text{and} \quad Q_2 = \{\{x_1, x_2, x_3, x_5\}\}$$

In both models, the set of applicable (and applied) defeasible rules is maximal; in contrast to the AF- and ADF-based semantics, there is no third model in which no defeasible rule has been applied.

We consider this eagerness to apply defeasible rules one of the most important differences between our direct semantics and the several previously seen translation-based semantics. As another difference, the outcome (model) of the possible-worlds semantics is not a propositional valuation, but a propositional theory (the set of all propositional formulas that are true in all worlds that are considered possible by the epistemic state). With respect to consistency of this theory, we note that, given a defeasible theory base TB, there are essentially two possibilities for its possible-worlds semantics:

1. TB has the empty epistemic state as its only model;
2. TB has a non-empty model.

The first case is an indication of inconsistency on the level of strict rules.

Example 7 (Continued). Recall the defeasible theory base comprising $Lit = \{x, \neg x\}$, $StrInf = \{r_1 :\rightarrow x,\ r_2 :\rightarrow \neg x\}$ and $DefInf = \emptyset$. We see that none of the possible worlds $\emptyset$ and $\{x\}$ satisfies *both* strict rules. Thus the rule base has the model $\emptyset$ where no world is possible and its inconsistency is obviated.

For each model of a defeasible theory base, we have by definition that all its possible worlds satisfy the material implications associated with the strict rules. Thus, closure holds in each possible world and in particular in the propositional theory derived from the epistemic state.

So consistency and closure do not pose problems for the possible-worlds semantics. However, it has its issues with positive cyclic dependencies.

Example 5 (Continued). Recall the example saying that rain and wet grass usually accompany each other: $\mathit{DefInf} = \{r_1 : \mathit{rain} \Rightarrow \mathit{wet}, r_2 : \mathit{wet} \Rightarrow \mathit{rain}\}$. The theory base has two models, $Q_1 = W_A$ and $Q_2 = \{\{\mathit{rain}, \mathit{wet}\}\}$. In Q_1 nothing is known about rain or wet grass; in Q_2 both are known, where each is defeasibly derived from the other.

Such issues, which are problematic with regard to causality, motivate us to define a refined version of the model semantics that excludes such cycles, a *stable model semantics*. Roughly, for a model to be stable, there must be a constructive derivation of its defeasible conclusions. For instance, Q_2 above is not stable since the two conclusions cyclicly depend on each other.

To achieve this constructiveness technically, we need a refinement of the satisfaction relation for defeasible rules and the notion of a model. The key change is not to check satisfaction of a rule's body against the model itself, but to check that all defeasible conclusions can be derived either from strict knowledge or from defeasible conclusions that are themselves constructively derived. This intuition comes from similar constructions in logic programming and default logic. The more technical description is to try to reconstruct a given model in an acyclic way. This construction starts with the set W_A of all possible worlds. There, nothing is known because any world is considered possible. The construction now stepwise removes worlds that are no longer considered possible. The worlds violating some strict rules are the first to go. If this leads to an increase in knowledge, then defeasible rules might become applicable and are applied through the refined model relation. If this leads to a further increase in knowledge (that is, a further decrease in the set of possible worlds), then the process continues. Otherwise the process stops, in which case we check what has been constructed. If the model could be fully reconstructed, then it is stable, otherwise it is not.

Definition 8. Let TB be a defeasible theory base over a vocabulary A, $w \in W_A$ and $Q, R \subseteq W_A$.

$$
\begin{aligned}
Q, R, w \models r : \phi_1, \ldots, \phi_n \to \psi \text{ iff } & w \models (\phi_1 \wedge \ldots \wedge \phi_n) \supset \psi \text{ in propositional logic} \\
Q, R, w \models r : \phi_1, \ldots, \phi_n \Rightarrow \psi \text{ iff } & \text{there is a } v \in R \text{ and } 1 \leq i \leq n \text{ with } v \not\models \phi_i \\
& \text{or for all } v \in Q \text{ we have } v \not\models \psi \\
& \text{or } w \models \psi \\
Q, R, w \models \mathit{TB} \text{ iff } & Q, R, w \models r \text{ for all } r \in \mathit{StrInf} \cup \mathit{DefInf}
\end{aligned}
$$

Now set $R_0 \stackrel{\text{def}}{=} W_A$ and for $i \geq 0$ define

$$R_{i+1} \stackrel{\text{def}}{=} \{w \in W_A \mid Q, R_i, w \models TB\} \quad \text{and} \quad R_\infty \stackrel{\text{def}}{=} \bigcap_{i \geq 0} R_i$$

A set Q of possible worlds is a *stable model for* TB iff $Q = R_\infty$.

It can be shown that the name "stable *model*" is well-chosen in that every stable model is a model [10].[10]

Example 5 (Continued). Let us check if $Q_2 = \{\{rain, wet\}\}$ is stable. We initialise the set of possible worlds $R_0 = W_A = \{\emptyset, \{rain\}, \{wet\}, \{rain, wet\}\}$. Now for obtaining R_1 according to Definition 8, we observe that neither defeasible rule's premise is known in the epistemic state R_0 and we have $Q_2, R_0, w \models TB$ for every world $w \in W_A$. Thus $R_1 = R_0 = W_A$, we could not reconstruct Q_2 and therefore it is not a stable model.

For $Q_1 = W_A$, on the other hand, the process terminates likewise after the first step. In this case, Q_1 could be reconstructed and is thus stable.

While the stable model semantics can deal with defeasible cycles, it is at a loss with respect to strict cycles, that is, positive cyclic dependencies among literals in strict rules.

Example 6 (Continued). Recall that the only rules of this example are strict, and given by $StrInf = \{r_1 : x_1 \to x_2, \quad r_2 : x_2 \to x_1\}$. Since there are no defeasible rules, models and stable models coincide. Clearly any world satisfying both rules satisfies the propositional formula $x_1 \equiv x_2$, so the (stable) models of the theory base – there are two of them, $\{\emptyset\}$ and $\{\{x_1, x_2\}\}$ – correspond one-to-one to the models of the formula – $\emptyset$ and $\{x_1, x_2\}$. The second (stable) model, $\{\{x_1, x_2\}\}$, where we know that both atoms are true, might be undesired in a causal context such as that of Example 4.

Here, our alternative intuition (DR) for strict rules comes into play. It is closer to the intuition behind defeasible rules and basically says that a strict rule is a directed inference rule on the knowledge level, and so we can use the same techniques for breaking strict cycles that we used earlier for defeasible ones. The formal definition simply says that with epistemic state Q, R (definitely possible worlds Q, potentially possible worlds R) in actual world w, a strict rule is satisfied if and only if knowing the truth of the premises implies the actual truth of its conclusion, where "knowing" refers to the conservative knowledge estimate given by the potentially possible worlds R:

$$Q, R, w \models r : \phi_1, \ldots, \phi_n \to \psi \text{ iff there is a } v \in R \text{ and } 1 \leq i \leq n \text{ with } v \not\models \phi_i \text{ or } w \models \psi$$

[10] Roughly, for $i \geq 0$ we have $R_i \supseteq R_{i+1}$ whence for each stable model we have $Q = R_i$ for some $i \in \mathbb{N}$, furthermore it can be shown that $Q, Q, w \models r$ (Def. 8) iff $Q, w \models r$ (Def. 7).

The remaining definitions, in particular those of models and stable models, stay the same.[11] This formal semantics of strict rules is just like that for defeasible rules, only without the additional condition that checks that the conclusion is not known to be false.

With this alternative semantics for strict rules, also positive strict cycles can be treated by the stable model semantics. However, there is another problem: this semantics is not able to produce the desired outcome of the "Married John" rule base.

Example 2 (Continued). For the married John example and the intuition where strict rules are interpreted according to propositional material implication, we had two models, $Q_1 = \{\{g, w, b\}\}$ and $Q_2 = \{\{g, w, m, h\}\}$. Unfortunately, neither of the models persists when strict rules are interpreted according to our alternative intuition, where they are much closer to defeasible rules. Then, the semantics' eagerness to apply rules also applies to strict rules and leads to global inconsistency in the sense of allowing as the only model of the theory base the empty epistemic state. We exemplify this by showing that Q_1 is not a model any more: Recall that Q_1 is a model iff $Q_1 = \{w \in W_A \mid Q_1, w \models TB\}$, in other words, if and only if $\{g, w, b\}$ is the one single world v for which we find $Q_1, v \models TB$. However, this is not the case. There is another world, $v' = \{g, w, b, m\}$, which satisfies the theory base in the epistemic state Q_1: First of all, the two strict rules r_1 and r_2 are satisfied by Q_1, v' since $v' \models w$ and $v' \models g$. We also have $Q_1, v' \models r_3 : b \to \neg h$ since $v' \models \neg h$. We find that $Q_1, v' \models r_4 : m \to h$ since $Q_1 \not\models m$. Finally, we can also show that $Q_1, v' \models r_5 : w \Rightarrow m$ because $v' \models m$; and that $Q_1, v' \models r_6 : g \Rightarrow b$ since $v' \models b$.

The problem is caused by r_4. Roughly speaking, there is incomplete knowledge about m – it is not known although it holds. In general, it is clear that the world v' should not be considered possible since in it John is a married bachelor. But the way strict rules are interpreted according to the alternative intution, the semantics has no way to figure this out, because strict rules do not operate on the level of single worlds, but only through the interaction of epistemic states and single worlds. A similar thing happens for Q_2; likewise it can be verified that the theory has no models at all.

This illustrates the difficulty of devising a semantics for defeasible theory bases that both possesses an eagerness to apply rules as well as it prevents self-supporting conclusions. Furthermore, our formalisation made it clear that the issue is linked to the question on which level strict rules should be enforced – on the level of single possible worlds or on theory level (knowledge level).

5.1. Relationship to Autoepistemic Logic

To explain the connection to related work in nonmonotonic reasoning, we briefly sketch how our possible-worlds semantics links to Moore's autoepistemic

[11] For the definition of a model the (DR) intuition uses the fact that $Q, w \models r$ iff $Q, Q, w \models r$.

logic (AEL) [16]. Propositional AEL enhances classical propositional logic by a unary modal connective $\mathbf{K}$ for knowledge. So for a formula φ, the AEL formula $\mathbf{K}\varphi$ stands for "φ is known." The semantics of autoepistemic logic is defined as follows: For a set B of formulas (the initial beliefs), a set T is an *expansion* of B if it coincides with the deductive closure of $B \cup \{\mathbf{K}\varphi \mid \varphi \in T\} \cup \{\neg\mathbf{K}\varphi \mid \varphi \notin T\}$. In words, T is an expansion if it equals what can be derived using the initial beliefs B and positive and negative introspection with respect to T itself.[12] The intuition behind $\mathbf{K}$ can be used to define a straightforward translation from theory bases into autoepistemic logic for the intuition (MI) behind strict rules.

Definition 9. Let $TB = (\mathit{Lit}, \mathit{StrInf}, \mathit{DefInf})$ be a defeasible theory. Define an autoepistemic theory $\Omega(TB)$ as follows.

$$\begin{aligned} \Omega(TB) &\stackrel{\text{def}}{=} \{\Omega(r) \mid r \in \mathit{StrInf} \cup \mathit{DefInf}\} \\ \Omega(\phi_1, \ldots, \phi_n \to \psi) &\stackrel{\text{def}}{=} (\phi_1 \wedge \ldots \wedge \phi_n) \supset \psi \\ \Omega(\phi_1, \ldots, \phi_n \Rightarrow \psi) &\stackrel{\text{def}}{=} (\mathbf{K}(\phi_1 \wedge \ldots \wedge \phi_n) \wedge \neg\mathbf{K}\neg\psi) \supset \psi \end{aligned}$$

With this translation, theory base models according to Definition 7 correspond one-to-one to expansions of the resulting autoepistemic theory. Likewise, stable models of the theory base are in one-to-one correspondence with strong expansions of the autoepistemic theory, a constructive refinement of the original expansion semantics [10].

For our alternative intuition (DR) for strict rules, the associated AEL translation is $\Omega(\phi_1, \ldots, \phi_n \to \psi) \stackrel{\text{def}}{=} (\mathbf{K}(\phi_1 \wedge \ldots \wedge \phi_n)) \supset \psi$. It is readily seen that this translation is quite close to that of a defeasible rule. The relation of our intuition behind theory bases with default logic [18] is immediate from reversing Konolige's translation [14], but we cannot give more details for a lack of space.

5.2. Defining Further Semantics

The translation from defeasible theory bases into autoepistemic logic immediately provides us with the possibility to define further argumentation semantics in terms of possible-world structures. Up to now, we explicitly only considered possible-world structures $Q \subseteq W_A$ that were in a sense two-valued, that is, a possible world $w \in W_A$ was either considered an epistemic alternative ($w \in Q$) or not ($w \notin Q$). This is alike to the stable semantics in argumentation, where each argument is either accepted or rejected. Of course, there are also three-valued argumentation semantics, like the complete semantics, where the status of an argument might be neither accepted nor rejected, but undecided. To generalise such three-valued semantics to a possible-world setting, we need possible-world structures in which the epistemic status of a possible world w can be likewise undecided, that is, for all that we know, the world w *might* be an epistemic alternative.

[12] Moore himself also gave a possible-worlds based treatment of autoepistemic logic [10], which was an inspiration for this work.

Denecker et al. [10] provided such a three-valued (even four-valued) possible-world treatment for autoepistemic logic. This treatment is embedded into the general algebraic framework of *approximation fixpoint theory* [9]. There, knowledge bases are associated with certain operators, and the semantics of the knowledge bases is then defined via fixpoints of these operators. In previous work of our own, we generalised several argumentation semantics to this abstract, operator-based setting [19]. Applying these general definitions of semantics to the approximation operator for autoepistemic logic as defined by Denecker et al. [10] immediately yields all of these semantics for defeasible theory bases. The precise technical definitions are straightforward to obtain and we omit them here for a lack of space. We rather give some examples to provide a glimpse of how some of the generalised semantics behave.

First of all, we want to note that most semantics for argumentation frameworks allow for more than one generalisation. We have seen this already in the case of the stable extension semantics, which can be generalised to ADFs in at least two ways, to models and *stable* models. Likewise, we presented two versions of two-valued epistemic semantics for defeasible theory bases. In the same vein, the grounded semantics for abstract argumentation can be generalised in at least two ways: to the Kripke-Kleene semantics, the cycle-supporting version of the grounded semantics, and to the well-founded semantics, the cycle-rejecting version of the grounded semantics [10, 19].

Let us consider the rain/wet grass example (Example 5). There, the grounded (Kripke Kleene) semantics considers the world $\{rain, wet\}$ to be definitely possible, and all other worlds to be *potentially* possible. The grounded (well-founded) semantics for this example corresponds to the two-valued epistemic model given by W_A and considers all worlds to be definitely possible. Intuitively, the well-founded semantics does not derive any knowledge from the two mutually supporting defeasible rules (all possible worlds occur in all stable models), while the Kripke-Kleene semantics lends some more credence to the world where both rain and wet grass are true (because this one world occurs in both models, while all others only occur in one of them).

Example 9. Consider the literals $Lit = \{x_1, x_2, \neg x_1, \neg x_2\}$ and the theory base given by defeasible rules $DefInf = \{r_1 :\Rightarrow x_1, \quad r_2 :\Rightarrow \neg x_1\}$ and the strict rule $StrInf = \{r_3 : x_1 \to x_2\}$. It is clear that not both defeasible rules can be applied, so there are two different models: $Q_1 = \{\{x_1, x_2\}\}$ where r_1 has been applied, and r_3 then infers x_2; and $Q_2 = \{\{\}, \{x_2\}\}$ where r_2 has been applied, and we do not know about x_2. In (both versions of) the grounded semantics of this theory base, no world is definitely considered possible, as the two models are disjoint. However, all the worlds in $Q_1 \cup Q_2$ are considered potentially possible.

Likewise, in Example 3, both versions of grounded semantics consider the three worlds $\{x_1, x_2, x_3\}$, $\{x_1, x_2, x_3, x_4\}$ and $\{x_1, x_2, x_3, x_5\}$ to be possible, albeit none of them definitely so. This shows that the generalised grounded semantics are not equal to sceptical reasoning among (stable) models, but rather an independent, weaker semantics. Indeed, this and other relationships between argumentation semantics carry over to their generalised versions [19].

6. Conclusion

We presented a translation from theory bases to abstract dialectical frameworks. The translated frameworks satisfy the rationality postulates closure and direct/indirect consistency, which we generalised to make them independent of a specific target formalism. Furthermore, the translated frameworks can detect inconsistencies in the rule base and cyclic supports amongst literals. We also showed that the translation involves at most a quadratic blowup and is therefore effectively computable. In addition, our translation produces a number of statements which is linear in the size of the theory base and can be considered efficient in this regard. (In the approach of [6] the number of produced arguments is unbounded in general.) In terms of desired behaviour, we compared our translation to previous approaches from the literature [6, 23, 24] and demonstrated how we avoid common problems. We also introduced possible-worlds semantics as a language to "think aloud" about defeasible theories. Along with this we presented two possible intuitions for strict rules and argued why we prefer one over the other. Of course, other intuitions are possible, and we mainly consider the present work a start for formulating intuitions in a formally precise way.

In earlier work, Brewka and Gordon [3] translated Carneades [13] argument evaluation structures (directly) to ADFs. They extended the original Carneades formalism by allowing cyclic dependencies among arguments. Meanwhile, Van Gijzel and Prakken [22] also translated Carneades into AFs (via ASPIC+ [15], that extends and generalises the definitions of Caminada and Amgoud [6]). They can deal with cycles, but there is only one unique grounded, preferred, complete, stable extension. Thus the semantic richness of abstract argumentation is not used, and the user cannot choose whether they want to accept or reject circular justifications of arguments. In contrast, in the approach of Brewka and Gordon [3] the user can decide whether cyclic justifications should be allowed or disallowed, by choosing models or *stable* models as ADF semantics.

We regard this work as another piece of evidence that abstract dialectical frameworks are well-suited as target formalisms for translations from rule-based nonmonotonic languages such as theory bases. A natural next step is to consider as input the specification language of ASPIC+ [15], for which a recent approach to preferences between statements [5] is a good starting point. In view of possible semantics for defeasible theories, it also seems fruitful to look at additional rationality postulates, for example those studied by Caminada et al. [7] or Dung and Thang [12]. Further work could also encompass the study of further ADF semantics, like complete or preferred models [5], and whether our translation to ADFs can be modified such that it is eager to apply defeasible rules and even coincides with our possible-world semantics. Finally, we can compare existing approaches to cycles in AFs [1, 2] with the treatment of cycles in ADFs.

Acknowledgements. The author is grateful to Gerhard Brewka for informative discussions. He also thanks Leila Amgoud and Martin Caminada for clarifying some aspects of the ASPIC framework. Several anonymous referees provided constructive feedback on earlier versions of the paper. This research has been partially supported by DFG under project BR-1817/7-1.

References

[1] Pietro Baroni, Massimiliano Giacomin, and Giovanni Guida. SCC-recursiveness: A general schema for argumentation semantics. *Artificial Intelligence*, 168(1–2):162–210, 2005.

[2] Pietro Baroni, Paul E. Dunne, and Massimiliano Giacomin. On the resolution-based family of abstract argumentation semantics and its grounded instance. *Artificial Intelligence*, 175(3–4):791–813, 2011.

[3] Gerhard Brewka and Thomas F. Gordon. Carneades and Abstract Dialectical Frameworks: A Reconstruction. In *Computational Models of Argument: Proceedings of COMMA 2010*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 3–12. IOS Press, September 2010.

[4] Gerhard Brewka and Stefan Woltran. Abstract Dialectical Frameworks. In *Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, pages 102–111, 2010.

[5] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes Peter Wallner, and Stefan Woltran. Abstract Dialectical Frameworks Revisited. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 803–809. IJCAI/AAAI, August 2013.

[6] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5–6):286–310, 2007.

[7] Martin W.A. Caminada, Walter A. Carnielli, and Paul E. Dunne. Semi-stable Semantics. *Journal of Logic and Computation*, 22(5):1207–1254, 2012.

[8] Marc Denecker, D. Theseider-Dupré, and Kristof Van Belleghem. An Inductive Definition Approach to Ramifications. *Linköping Electronic Articles in Computer and Information Science*, 3(7):1–43, January 1998.

[9] Marc Denecker, Victor Marek, and Miroslaw Truszczyński. Approximations, Stable Operators, Well-Founded Fixpoints and Applications in Nonmonotonic Reasoning. In *Logic-Based Artificial Intelligence*, pages 127–144. Kluwer Academic Publishers, 2000.

[10] Marc Denecker, V. Wiktor Marek, and Miroslaw Truszczyński. Uniform Semantic Treatment of Default and Autoepistemic Logics. *Artificial Intelligence*, 143(1):79–122, 2003.

[11] Phan Minh Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. *Artificial Intelligence*, 77:321–358, 1995.

[12] Phan Minh Dung and Phan Minh Thang. Closure and consistency in logic-associated argumentation. *Journal of Artificial Intelligence Research*, 49: 79–109, 2014.

[13] Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10–15): 875–896, 2007.

[14] Kurt Konolige. On the Relation Between Default and Autoepistemic Logic. *Artificial Intelligence*, 35(3):343–382, 1988.

[15] Sanjay Modgil and Henry Prakken. A general account of argumentation and preferences. *Artificial Intelligence*, 195(0):361–397, 2013.

[16] Robert Moore. Semantical Considerations of Nonmonotonic Logic. *Artificial Intelligence*, 25(1):75–94, 1985.

[17] John L Pollock. Defeasible reasoning. *Cognitive Science*, 11(4):481–518, 1987.

[18] Raymond Reiter. A Logic for Default Reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[19] Hannes Strass. Approximating operators and semantics for abstract dialectical frameworks. *Artificial Intelligence*, 205:39–70, December 2013.

[20] Hannes Strass. Instantiating knowledge bases in Abstract Dialectical Frameworks. In *Proceedings of the Fourteenth International Workshop on Computational Logic in Multi-Agent Systems (CLIMA XIV)*, volume 8143 of *LNCS*, pages 86–101. Springer, September 2013.

[21] Hannes Strass and Johannes Peter Wallner. Analyzing the Computational Complexity of Abstract Dialectical Frameworks via Approximation Fixpoint Theory. In *Proceedings of the Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, pages 101–110, Vienna, Austria, July 2014.

[22] Bas Van Gijzel and Henry Prakken. Relating Carneades with abstract argumentation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence – Volume Two*, pages 1113–1119. IJCAI/AAAI, 2011.

[23] Adam Wyner, Trevor Bench-Capon, and Paul Dunne. Instantiating knowledge bases in abstract argumentation frameworks. In *Proceedings of the AAAI Fall Symposium – The Uses of Computational Argumentation*, 2009.

[24] Adam Wyner, Trevor J. M. Bench-Capon, and Paul E. Dunne. On the instantiation of knowledge bases in abstract argumentation frameworks. In *Proceedings of CLIMA XIV*, volume 8143 of *LNAI*, pages 34–50. Springer, September 2013.