Analyzing the Computational Complexity of Abstract Dialectical Frameworks via Approximation Fixpoint Theory

Hannes Strass[a,*], Johannes Peter Wallner[b]

[a] *Computer Science Institute, Leipzig University*
Augustusplatz 10, 04109 Leipzig, Germany
[b] *Institute of Information Systems, Vienna University of Technology*
Favoritenstraße 9-11, 1040 Vienna, Austria

Abstract

Abstract dialectical frameworks (ADFs) have recently been proposed as a versatile generalization of Dung's abstract argumentation frameworks (AFs). In this paper, we present a comprehensive analysis of the computational complexity of ADFs. Our results show that while ADFs are one level up in the polynomial hierarchy compared to AFs, there is a useful subclass of ADFs which is as complex as AFs while arguably offering more modeling capacities. As a technical vehicle, we employ the approximation fixpoint theory of Denecker, Marek and Truszczyński, thus showing that it is also a useful tool for complexity analysis of operator-based semantics.

Keywords: abstract dialectical frameworks, computational complexity, approximation fixpoint theory

1. Introduction

Formal models of argumentation are increasingly being recognized as viable tools in knowledge representation and reasoning [5]. A particularly popular formalism are Dung's abstract argumentation frameworks (AFs) [24]. AFs treat arguments as abstract entities and natively represent only attacks between them using a binary relation. Typically, abstract argumentation frameworks are used as a target language for translations from more concrete languages. For example, the Carneades formalism for structured argumentation [35] has been translated to AFs [45]; Caminada and Amgoud [13] and Wyner et al. [47] translate rule-based defeasible theories into AFs. Despite their popularity, abstract argumentation frameworks have limitations. Most significantly, their limited modeling capacities are a notable obstacle for applications: arguments can only attack one another. Furthermore, Caminada and Amgoud [13] observed how AFs that arise as translations of defeasible theories sometimes lead to unintuitive conclusions.

To address the limitations of abstract argumentation frameworks, researchers have proposed quite a number of generalizations of AFs [12]. Among the most general of those are Brewka and Woltran's *abstract dialectical frameworks (ADFs)* [9]. ADFs are even more abstract than AFs: while in AFs arguments are abstract and the relation between arguments is fixed to attack,

*Corresponding author
Email addresses: `strass@informatik.uni-leipzig.de` (Hannes Strass), `wallner@dbai.tuwien.ac.at` (Johannes Peter Wallner)

in ADFs also the relations are abstract (and called *links*). The relationship between different arguments (called *statements* in ADFs) is specified by *acceptance conditions*. These are Boolean functions indicating the conditions under which a statement s can be accepted when given the acceptance status of all statements with a direct link to s (its *parents*). ADFs have been successfully employed to address the shortcomings of AFs: Brewka and Gordon [8] translated Carneades to ADFs and for the first time allowed cyclic dependencies amongst arguments; for rule-based defeasible theories we [41, 43] showed how ADFs can be used to deal with the problems observed by Caminada and Amgoud [13].

There is a great number of semantics for AFs already, and many of them have been generalized to ADFs. Thus it might not be clear to potential ADF users which semantics are adequate for a particular application domain. In this regard, knowing the computational complexity of semantics can be a valuable guide. However, existing complexity results for ADFs are scattered over different papers, miss several semantics and some of them present upper bounds only. In this paper, we provide a comprehensive complexity analysis for ADFs. In line with the literature, we represent acceptance conditions by propositional formulas as they provide a compact and elegant way to represent Boolean functions.

Technically, we base our complexity analysis on the approximation fixpoint theory (AFT) by Denecker, Marek and Truszczyński [18, 19, 20]. This powerful framework provides an algebraic account of how monotone and nonmonotone two-valued operators can be approximated by monotone three- or four-valued operators. (As an example of an operator to be approximated, think of the two-valued van Emden-Kowalski consequence operator from logic programming.) AFT embodies the intuitions of decades of KR research; we believe that this is very valuable also for relatively recent languages (such as ADFs), because we get the enormously influential formalizations of intuitions of Reiter and others for free. (As a liberal variation on Newton, we could say that approximation fixpoint theory allows us to take the elevator up to the shoulders of giants instead of walking up the stairs.) In fact, approximation fixpoint theory can be and partially has already been used to define some of the semantics of ADFs [11, 40]. There, we generalized various AF and logic programming semantics to ADFs using AFT, which has provided us with two families of semantics, that we call – for reasons that will become clear later – *approximate* and *ultimate*, respectively. Intuitively speaking, both families approximate the original two-valued model semantics of ADFs, where the ultimate family is more *precise* in a formally defined sense. The present paper employs approximating operators for complexity analysis and thus shows that AFT is also well-suited for studying the computational complexity of formalisms.

Along with providing a comparison of the approximate and ultimate families of semantics, our main results can be summarized as follows. We show that: (1) the computational complexity of ADF decision problems is one level up in the polynomial hierarchy from their AF counterparts [28]; (2) the ultimate semantics are almost always as complex as the approximate semantics, with the notable exceptions of two-valued stable models, and conflict-free and naive semantics; (3) there is a certain subclass of ADFs, called *bipolar* ADFs (BADFs), which is of the same complexity as AFs, with the single exception of skeptical reasoning for naive semantics. Intuitively, in bipolar ADFs all links between statements are *supporting* or *attacking*. To formalize these notions, Brewka and Woltran [9] gave a precise semantical definition of support and attack. In our work, we assume that the link types are specified by the user along with the ADF. We consider this a harmless assumption since the existing applications of ADFs produce bipolar ADFs where the link types are known [8, 41]. This attractiveness of bipolar ADFs from a KR point of view is the most significant result of the paper: it shows that BADFs offer – in addition to AF-like and more general notions of attack – also syntactical notions of support *without any increase in computational cost*.

In BADFs, support for a statement s can be anything among "set support" (all statements in a certain set must be accepted for the support to become active) or "individual support" (at least one statement supporting s must be accepted for the support to become active). In the same vein, BADFs offer "set attack" (all statements in a certain set must be accepted for the attack to become active) and the traditional "individual attack" known from AFs (at least one statement attacking s must be accepted for the attack to become active). Naturally, in BADFs all these different notions of support and attack can be freely combined.

Previously, Brewka et al. [10] translated BADFs into AFs for two-valued semantics and suggested indirectly that the complexities align.[1] Here we go a direct route, which has more practical relevance since it immediately affects algorithm design. Our work was also inspired by the complexity analysis of assumption-based argumentation by Dimopoulos et al. [23] – they derived generic results in a way similar to ours.

Our complexity results aligning AFs and BADFs are especially remarkable with regard to expressiveness in the model-theoretic sense. While it remains elusive what kinds of sets of two-valued interpretations the class of AFs can express exactly [4], we know that even bipolar ADFs can express strictly more than that (at least all $\subseteq$-antichains), and general (non-bipolar) ADFs can express any set of two-valued interpretations with the two-valued model semantics [42]. This shows that AFs (under stable extension/labelling semantics) – while being of equal computational complexity – are strictly less expressive than ADFs (under model semantics, one of the ADF counterparts of AF stable semantics).

We also provide results on the existence of certain types of interpretations in a general setting. For example, it follows from our results that any approximating operator in a complete partial order always possesses preferred and naive interpretations. This generalizes a corresponding result by Dung [24] about the existence of AF preferred extensions to finite and infinite ADFs, logic programs, default theories, and beyond [19]. The conflict-free (and naive) semantics that we consider here is – strictly speaking – also a novel contribution of this paper, as previous definitions of conflict-freeness were either two-valued [9] or direct generalizations of the corresponding three-valued AF notion [40]. The definition we use here is simpler, more intuitive and still a generalization of AFs' conflict-free sets.

One important proof technique of this paper is to employ ADFs' acceptance conditions' representation via propositional formulas and to partially evaluate them. For a propositional formula φ over vocabulary P and $X \subseteq Y \subseteq P$ we define the *partial valuation of* φ *by* (X, Y) as

$$\varphi^{(X,Y)} = \varphi[p/\mathbf{t} : p \in X][p/\mathbf{f} : p \in P \setminus Y]$$

Intuitively, the pair (X, Y) represents a partial interpretation of P where all elements of X are true and all elements of $P \setminus Y$ are false.[2] The partial evaluation of φ with (X, Y) takes the two-valued part of (X, Y) and replaces the evaluated variables by their truth values. Naturally, $\varphi^{(X,Y)}$ is a formula over the vocabulary $Y \setminus X$, that is, only contains variables that have no classical truth value (true or false) in the pair (X, Y). In particular, for any total interpretation (X, X), the partial evaluation $\varphi^{(X,X)}$ is a Boolean expression consisting only of truth constants and connectives and thus has a fixed truth value (either true or false).

We will show that approximate and ultimate ADF operators (and thus all of the operator-based ADF semantics) can be defined in terms of partial evaluations of acceptance formulas. For example, in the new three-valued conflict-free semantics that we introduce, a statement s can only be set to true in an interpretation (X, Y) if the partial evaluation of its acceptance formula with the interpretation – the formula $\varphi_s^{(X,Y)}$ – is satisfiable. Symmetrically, s can only

[1] Additionally, in contrast to Brewka et al. [10], we use a revised version of the stable model semantics [40, 11].

[2] Equivalently, the pair (X, Y) represents a three-valued interpretation where all elements of $Y \setminus X$ are undefined.

be set to false in (X,Y) if $\varphi_s^{(X,Y)}$ is refutable. For the three-valued admissible semantics, the justification standards are higher. There, setting s to true is only justified if $\varphi_s^{(X,Y)}$ is irrefutable (a tautology), setting s to false is only justified if $\varphi_s^{(X,Y)}$ is unsatisfiable. This logical view of (argumentation) semantics thus provides a novel perspective on different, graded notions of acceptability.

The paper proceeds as follows. We first provide the background on approximation fixpoint theory, abstract dialectical frameworks and the necessary elements of complexity theory. In the section afterwards, we define the relevant decision problems, survey existing complexity results, use examples to illustrate how operators revise ADF interpretations and show generic upper complexity bounds along with some other useful preparatory technical results. In the main section on complexity results for general ADFs, we back up the upper bounds with matching lower bounds; the section afterwards does the same for bipolar ADFs. We end with a brief discussion of related and future work. This paper is a revised and extended version of [44].

2. Background

A *complete lattice* is a partially ordered set (poset) $(L, \sqsubseteq)$ where every subset S of L has a least upper bound $\bigsqcup S \in L$ and a greatest lower bound $\bigsqcap S \in L$. In particular, a complete lattice has a least $(\bot)$ and a greatest $(\top)$ element.[3] An operator $O : L \to L$ is *monotone* if for all $x \sqsubseteq y$ we find $O(x) \sqsubseteq O(y)$. An $x \in L$ is a *fixpoint* of O if $O(x) = x$; an $x \in L$ is a *prefixpoint* of O if $O(x) \sqsubseteq x$ and a *postfixpoint* of O if $x \sqsubseteq O(x)$. Due to a fundamental result by Tarski and Knaster, for any monotone operator O on a complete lattice, the set of its fixpoints forms a complete lattice itself [17, Theorem 2.35]. In particular, its least fixpoint $\mathit{lfp}(O)$ exists.

In this paper, we will be concerned with more general algebraic structures: complete partially ordered sets (CPOs). A CPO is a partially ordered set $(C, \leq)$ with a $\leq$-least element where each directed subset $D \subseteq C$ has a least upper bound $\bigsqcup D \in C$. A set is directed iff it is nonempty and each pair of elements has an upper bound in the set. Clearly every complete lattice is a complete partially ordered set, but not necessarily vice versa. Fortunately, complete partially ordered sets still guarantee the existence of (least) fixpoints for monotone operators.

Theorem 2.1 ([17, Theorem 8.22])**.** *In a complete partially ordered set $(C, \leq)$, any $\leq$-monotone operator $O : C \to C$ has a least fixpoint.*

2.1. Approximation Fixpoint Theory

Denecker et al. [18] introduce the important concept of an approximation of an operator. In the study of semantics of knowledge representation formalisms, elements of lattices represent objects of interest. Operators on lattices transform such objects into others according to the contents of some knowledge base. Consequently, fixpoints of such operators are then objects that are fully updated – informally, the knowledge base can neither increase nor decrease the amount of information in a fixpoint.

To study fixpoints of operators O, Denecker et al. study their *approximation operators* $\mathcal{O}$.[4] When O operates on a set L, its approximation $\mathcal{O}$ operates on pairs $(x, y) \in L \times L$. Such a pair (x, y) can be seen as representing a *set* of lattice elements by providing a lower bound x and an upper bound y. Consequently, (x, y) approximates all $z \in L$ such that $x \sqsubseteq z \sqsubseteq y$. We will

[3] When dealing with different structures at the same time, we sometimes index $\bigsqcup, \bigsqcap, \bot, \top$ to indicate to which structure they belong. For example, $\bot_L$ refers to the $\sqsubseteq$-least element of the lattice $(L, \sqsubseteq)$.

[4] The approximation of an operator O is typographically indicated by a calligraphic $\mathcal{O}$.

restrict our attention to *consistent* pairs – those where $x \sqsubseteq y$, that is, the set of approximated elements is nonempty; we denote the set of all consistent pairs over L by L^c. A pair (x, y) with $x = y$ is called *exact* – it "approximates" a single element of the original lattice.

It is natural to order approximating pairs according to their information content. Formally, for $x_1, x_2, y_1, y_2 \in L$ define the *information ordering*

$$(x_1, y_1) \leq_i (x_2, y_2) \text{ iff } x_1 \sqsubseteq x_2 \text{ and } y_2 \sqsubseteq y_1$$

This ordering and the restriction to consistent pairs leads to a complete partially ordered set $(L^c, \leq_i)$, the *consistent CPO*. For example, the *trivial pair* $(\bot, \top)$ consisting of $\sqsubseteq$-least $\bot$ and $\sqsubseteq$-greatest lattice element $\top$ approximates all lattice elements and thus contains no information – it is the least element of the CPO $(L^c, \leq_i)$; exact pairs (x, x) are the maximal elements of $(L^c, \leq_i)$.

To define an approximation operator $\mathcal{O} : L^c \to L^c$, one essentially has to define two functions: a function $\mathcal{O}' : L^c \to L$ that yields a revised *lower* bound (first component) for a given pair; and a function $\mathcal{O}'' : L^c \to L$ that yields a revised *upper* bound (second component) for a given pair. Accordingly, the overall approximation is then given by $\mathcal{O}(x, y) = (\mathcal{O}'(x, y), \mathcal{O}''(x, y))$ for $(x, y) \in L^c$. The operator $\mathcal{O} : L^c \to L^c$ is *approximating* iff it is $\leq_i$-monotone and it satisfies $\mathcal{O}'(x, x) = \mathcal{O}''(x, x)$ for all $x \in L$, that is, $\mathcal{O}$ assigns exact pairs to exact pairs. Such an $\mathcal{O}$ then *approximates* an operator $O : L \to L$ on the original lattice iff $\mathcal{O}'(x, x) = O(x)$ for all $x \in L$.

The main contribution of Denecker et al. [18] was the association of the *stable operator* to an approximating operator. Their original definition was four-valued; in this paper we are only interested in two-valued stable models and simplified the definitions. For an approximating operator $\mathcal{O}$ on a consistent CPO, a (two-valued) fixpoint $(x, x) \in L^c$ of $\mathcal{O}$ is a (two-valued) *stable model of* $\mathcal{O}$ iff x is the least fixpoint of the operator $\mathcal{O}'(\cdot, x)$ defined by $w \mapsto \mathcal{O}'(w, x)$ for $w \sqsubseteq x$. This general, lattice-theoretic approach yields a uniform treatment of the standard semantics of the major nonmonotonic knowledge representation formalisms – logic programming, default logic and autoepistemic logic [19].

In subsequent work, Denecker et al. [20] presented a general, abstract way to define the most precise – called the *ultimate* – approximation of a given operator O. Most precise here refers to a generalization of $\leq_i$ to operators, where for $\mathcal{O}_1, \mathcal{O}_2$, they define $\mathcal{O}_1 \leq_i \mathcal{O}_2$ iff for all $(x, y) \in L^c$ it holds that $\mathcal{O}_1(x, y) \leq_i \mathcal{O}_2(x, y)$. Denecker et al. [20] showed that the most precise approximation of O is $\mathcal{U}_O : L^c \to L^c$ with

$$\mathcal{U}_O(x, y) = \left(\bigsqcap \{ O(z) \mid x \sqsubseteq z \sqsubseteq y \}, \bigsqcup \{ O(z) \mid x \sqsubseteq z \sqsubseteq y \} \right)$$

where $\bigsqcap$ denotes the greatest lower bound and $\bigsqcup$ the least upper bound in the complete lattice $(L, \sqsubseteq)$.

2.2. Abstract Dialectical Frameworks

An abstract dialectical framework (ADF) is a directed graph whose nodes represent statements or positions which can be accepted or not. The links represent dependencies: the status of a node s only depends on the status of its parents (denoted $par(s)$), that is, the nodes with a direct link to s. In addition, each node s has an associated acceptance condition C_s specifying the exact conditions under which s is accepted. C_s is a function assigning to each subset of $par(s)$ one of the truth values $\mathbf{t}$, $\mathbf{f}$. Intuitively, if for some $R \subseteq par(s)$ we have $C_s(R) = \mathbf{t}$, then s will be accepted provided the nodes in R are accepted and those in $par(s) \setminus R$ are not accepted.

Definition 2.1. An *abstract dialectical framework* is a tuple $\Xi = (S, L, C)$ where

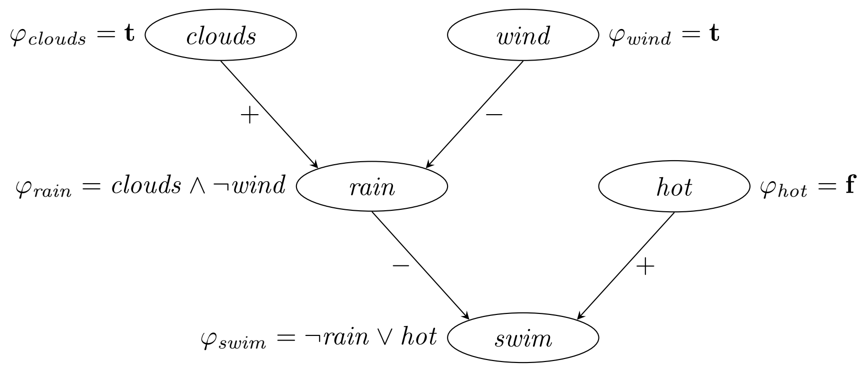
- S is a set of statements (positions, nodes),

- $L \subseteq S \times S$ is a set of links,
- $C = \{C_s\}_{s \in S}$ is a collection of total functions $C_s : 2^{par(s)} \to \{\mathbf{t}, \mathbf{f}\}$, one for each statement s. The function C_s is called *acceptance condition of* s.

It is often convenient to represent acceptance conditions by propositional formulas. In particular, we will do so for the complexity results of this paper. There, each C_s is represented by a propositional formula φ_s over $par(s)$. Then, clearly, $C_s(R \cap par(s)) = \mathbf{t}$ iff $R \models \varphi_s$. Furthermore, throughout the paper we will denote ADFs by Ξ and tacitly assume that $\Xi = (S, L, C)$ unless stated otherwise.

Brewka and Woltran [9] introduced a useful subclass of ADFs called *bipolar*: Intuitively, in bipolar ADFs (BADFs) each link is supporting or attacking (or both). Formally, a link $(r, s) \in L$ is *supporting in* Ξ iff for all $R \subseteq par(s)$, we have $C_s(R) = \mathbf{t}$ implies $C_s(R \cup \{r\}) = \mathbf{t}$; symmetrically, a link $(r, s) \in L$ is *attacking in* Ξ iff for all $R \subseteq par(s)$, we have $C_s(R \cup \{r\}) = \mathbf{t}$ implies $C_s(R) = \mathbf{t}$. An ADF $\Xi = (S, L, C)$ is *bipolar* iff all links in L are supporting or attacking; we use L^+ to denote all supporting and L^- to denote all attacking links of L in Ξ. For an $s \in S$ we define $att_\Xi(s) = \{x \mid (x, s) \in L^-\}$ and $supp_\Xi(s) = \{x \mid (x, s) \in L^+\}$. In this paper we assume that L^+ and L^- are given with a BADF, that is, the link types are known.

Example 2.1 (Adapted from [9, Example 6]). Consider a scenario where we want to decide whether we go for a *swim*. We do so if there is no *rain*, or it is *hot*. It is warm, but not hot, and there are *clouds* indicating that it might rain. However the reliable weather forecast predicts *wind* that will blow away the clouds. Using the vocabulary $S = \{clouds, wind, rain, hot, swim\}$, we devise the bipolar ADF $D_{swim} = (S, L^+ \cup L^-, C)$ shown below to model this deliberation process. Here, statements are depicted as nodes, edges represent links and acceptance conditions are written as propositional formulas next to the statements.



Supporting and attacking links are designated using the labels + and −; this is however only for illustration as the polarity of the links can be read off the acceptance formulas. The statement *rain*, for example, is supported by the statement *clouds* and attacked by the statement *wind*. According to φ_{rain}, the attack from *wind* is stronger than the support from *clouds*. That is, as soon as we accept *wind*, we must reject *rain*. On the other hand, *swim* is attacked by rain and supported by hot. Here, by φ_{swim}, the support from *hot* is stronger than the attack from *rain*; or put another way, the missing attack from *rain* is stronger than the missing support from *hot*. This effectively means that rejecting *rain* leads to accepting *swim*.

The semantics of ADFs can be defined using approximating operators. For two-valued semantics of ADFs we are interested in sets of statements, that is, we work in the complete lattice

$(A, \sqsubseteq) = (2^S, \subseteq)$. To approximate elements of this lattice, we use consistent pairs of sets of statements and the associated consistent CPO $(A^c, \leq_i)$ – the *consistent CPO over S-subset pairs.* Such a pair $(X, Y) \in A^c$ can be regarded as a three-valued interpretation where all elements in X are true, those in $Y \setminus X$ are unknown and those in $S \setminus Y$ are false. (This allows us to use "pair" and "interpretation" synonymously from now on.) The following definition specifies one particular way to revise a given three-valued interpretation.

Definition 2.2 ([40, Definition 3.1]). Let Ξ be an ADF. Define the operator $\mathcal{G}_\Xi : 2^S \times 2^S \to 2^S \times 2^S$ by

$$\mathcal{G}_\Xi(X, Y) = (\mathcal{G}'_\Xi(X, Y), \mathcal{G}'_\Xi(Y, X))$$
$$\mathcal{G}'_\Xi(X, Y) = \{s \in S \mid \exists B \subseteq par(s), C_s(B) = \mathbf{t}, B \subseteq X, (par(s) \setminus B) \cap Y = \emptyset\}$$

In a nutshell, statement s is included in the revised lower bound iff the input pair provides sufficient reason to do so, given acceptance condition C_s. To obtain some more intuition, it is instructive to look at the operator's behavior on consistent and inconsistent input pairs separately. Let Ξ be an ADF over statements S and let $X \subseteq Y \subseteq S$. Then (X, Y) is a consistent pair, and by definition, for $s \in S$, we have $s \in \mathcal{G}'_\Xi(X, Y)$ if and only if there is some $B \subseteq par(s)$ with $C_s(B) = \mathbf{t}$ (that is, $B \models \varphi_s$), $B \subseteq X$ and $(par(s) \setminus B) \cap Y = \emptyset$. We can think of this B as a two-valued interpretation of the parents of s. The last condition entails that s has no parents in $Y \setminus B$. Since $B \subseteq X$ this furthermore entails that s has no parents in $Y \setminus X$, that is, no parents that are undecided according to the pair (X, Y). But this means that the formula $\varphi_s^{(X,Y)}$ is a Boolean expression consisting only of truth constants and connectives. By $B \models \varphi_s$, the expression $\varphi_s^{(X,Y)}$ evaluates to true. For the converse pair (Y, X), which is not necessarily consistent, but still needed to compute a new upper bound, the reasoning is slightly more involved. Now we have $s \in \mathcal{G}'_\Xi(Y, X)$ if and only if there is some $B \subseteq par(s)$ with $B \models \varphi_s$, $B \subseteq Y$ and $(par(s) \setminus B) \cap X = \emptyset$. Again thinking of B as a two-valued interpretation of $par(s)$, the last condition entails that B must contain the true parents of s, that is, $par(s) \cap X \subseteq B$. Condition $B \subseteq Y$ means that any statement that is false in (X, Y) must be false in B. Altogether $s \in \mathcal{G}'_\Xi(Y, X)$ if and only if there is a two-valued interpretation B of $par(s)$ that evaluates φ_s to true and coincides with (X, Y) whenever (X, Y) assigns $\mathbf{t}$ or $\mathbf{f}$.

Although the operator is defined for all pairs (including inconsistent ones), its restriction to consistent pairs is well-defined since it maps consistent pairs to consistent pairs. This operator defines the *approximate* family of ADF semantics according to Table 1. Several of the abstract, operator-based semantics defined in Table 1 are quite recent, and inspired by semantics from logic programming and abstract argumentation [40].[5]

Based on the three-valued operator $\mathcal{G}_\Xi$, a two-valued one-step consequence operator for ADFs can be defined by $G_\Xi(X) = \mathcal{G}'_\Xi(X, X)$. Alternatively, for $\Xi = (S, L, C)$ we could specify

$$G_\Xi(X) = \{s \in S \mid X \models \varphi_s\}$$

The general result of Denecker et al. [20] (Theorem 5.6) then immediately defines the ultimate approximation of G_Ξ as the operator $\mathcal{U}_\Xi$ given by $\mathcal{U}_\Xi(X, Y) = (\mathcal{U}'_\Xi(X, Y), \mathcal{U}''_\Xi(X, Y))$ with

$$\mathcal{U}'_\Xi(X, Y) = \{s \in S \mid \text{ for all } Z \subseteq S \text{ with } X \subseteq Z \subseteq Y \text{ we have } Z \models \varphi_s\}$$
$$\mathcal{U}''_\Xi(X, Y) = \{s \in S \mid \text{ for some } Z \subseteq S \text{ with } X \subseteq Z \subseteq Y \text{ we have } Z \models \varphi_s\}$$

[5] To be precise, we used a slightly different technical setting there. The results can however be transferred to the present setting [20, Theorem 4.2].

Kripke-Kleene semantics	$lfp(\mathcal{O})$	grounded pair
conflict-free pair (x, y)	$x \sqsubseteq \mathcal{O}''(x, y)$ and $\mathcal{O}'(x, y) \sqsubseteq y$	conflict-free pair
M-conflict-free pair (x, y)	(x, y) is $\leq_i$-maximal conflict-free	naive pair
admissible/reliable pair (x, y)	$(x, y) \leq_i \mathcal{O}(x, y)$	admissible pair
three-valued supported model (x, y)	$(x, y) = \mathcal{O}(x, y)$	complete pair
M-supported model (x, y)	(x, y) is $\leq_i$-maximal admissible	preferred pair
two-valued supported model (x, x)	$(x, x) = \mathcal{O}(x, x)$	model
two-valued stable model (x, x)	$x = lfp(\mathcal{O}'(\cdot, x))$	stable model

Table 1: Operator-based semantical notions (and their argumentation names on the right) for a complete lattice $(L, \sqsubseteq)$ and an approximating operator $\mathcal{O} : L^c \to L^c$ on the consistent CPO $(L^c, \leq_i)$. While an approximating operator always possesses three-valued (post-)fixpoints, two-valued fixpoints need not exist. Clearly, any two-valued stable model is a two-valued supported model is a preferred pair is a complete pair is an admissible pair; furthermore the grounded pair is a complete pair. Any two-valued supported model is also a naive pair is a conflict-free pair.

Incidentally, Brewka and Woltran [9] already defined this operator, which was later used to define the *ultimate* family of ADF semantics according to Table 1 [11].[6] In this paper, we will refer to the two families of three-valued semantics as "approximate σ" and "ultimate σ" for σ among conflict-free, naive, admissible, grounded, complete, preferred and stable. For two-valued supported models (or simply models), approximate and ultimate semantics coincide (since the two approximating operators $\mathcal{G}_\Xi$ and $\mathcal{U}_\Xi$ approximate the same two-valued operator G_Ξ).

Our definition of conflict-free pairs differs from the one given in [40], but is still a valid generalization of the notion of conflict-free sets for AFs [24].[7] We chose this definition because it is symmetric and easier to work with. An AF $F = (A, R)$ is a pair with A a set and $R \subseteq A \times A$ a binary relation on A. A set $S \subseteq A$ is conflict-free in F if for all $a, b \in S$ it holds that $(a, b) \notin R$. The associated ADF of F is given by $\Xi = (A, R, C)$ with $\varphi_a = \bigwedge_{(b,a) \in R} \neg b$ for $a \in A$.

Proposition 2.2. *Let $F = (A, R)$ be an AF, Ξ be its associated ADF and $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$.*

1. *For each conflict-free set $X \subseteq A$, there exists $Y \subseteq A$ such that (X, Y) is a conflict-free pair of $\mathcal{O}$.*
2. *For each conflict-free pair (X, Y), its lower bound X is a conflict-free set.*

Proof. We make use of the fact that for any $P, Q \subseteq A$, we have $\mathcal{O}(P, Q) = (U_F(Q), U_F(P))$, which follows from [40, Proposition 4.1], where $U_F(S) = \{a \in A \mid S \text{ does not attack } a\}$ for $S \subseteq A$.

1. Let $X \subseteq A$ be conflict-free. Define $Y = U_F(X)$. Since X is conflict-free,

$$X \subseteq Y = U_F(X) = \mathcal{O}''(X, Y)$$

   Furthermore U_F is $\subseteq$-antimonotone, whence $X \subseteq U_F(X)$ implies

$$\mathcal{O}'(X, Y) = U_F(Y) = U_F(U_F(X)) \subseteq U_F(X) = Y$$

2. Let (X, Y) be a conflict-free pair. Then $X \subseteq \mathcal{O}''(X, Y) = U_F(X)$, whence X is a conflict-free set. □

[6] Technically, Brewka et al. [11] represented interpretations not by pairs $(X, Y) \in A^c$ but by mappings $v : S \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$ into the set of truth values $\mathbf{t}$ (true), $\mathbf{f}$ (false) and $\mathbf{u}$ (undefined or unknown or undecided). Clearly the two representations are interchangeable.

[7] Strictly speaking, our definition of conflict-free pairs is a new contribution of this paper. We present it in the background for readability.

Although Table 1 defines two-valued stable models also for the ultimate operator, Brewka et al. [11] have their own tailor-made definition of two-valued stable models. There, a two-valued model (M, M) is a *stable model* of an ADF $\Xi = (S, L, C)$ iff M is the lower bound of the ultimate grounded semantics of the reduced ADF $\Xi^M = (M, L \cap (M \times M), C^M)$ where the reduced acceptance formula for an $s \in M$ is given by the partial evaluation $\varphi_s^{(\emptyset,M)}$.[8] It is not hard to prove that the definition of two-valued stable models given by Brewka et al. [11] coincides with Denecker et al.'s ultimate two-valued stable models. We start with an easy observation.

Lemma 2.3. *Let φ be a propositional formula over vocabulary S, and let A, B, C, D be sets with $A \subseteq B \subseteq S$ and $C \subseteq D \subseteq S$.*

$$\left(\varphi^{(A,B)}\right)^{(C,D)} = \varphi^{(A\cup C, B\cap D)}$$

For the actual result (in particular for its proof), it is helpful to recall that the stable models of Brewka et al. [11] are models by definition.

Proposition 2.4. *The stable model semantics as defined by Brewka et al. [11] coincides with the ultimate two-valued stable model semantics of Denecker et al. [20].*

Proof. *Let $\Xi = (S, L, C)$ be an ADF and $M \subseteq S$ be a model of Ξ. We show that (M, M) is a Brewka et al.-stable-model of Ξ if and only if (M, M) is an ultimate two-valued stable model of Ξ. First, it is easy to see that M is the lower bound of the ultimate grounded semantics of the reduced ADF $\Xi^M = (M, L \cap (M \times M), C^M)$ if and only if (M, M) is the ultimate grounded semantics of Ξ^M. Furthermore, M is a model of Ξ, whence it is a model of Ξ^M. Thus all acceptance formulas in Ξ^M are satisfiable and for any $X \subseteq M$ we get $\mathcal{U}''_{\Xi^M}(X, M) = M$. That is, during computation of the least fixpoint of $\mathcal{U}_{\Xi^M}$, the upper bound remains constant at M. Now for any $X \subseteq M$ and $s \in S$, we have*

$$\begin{aligned} s \in \mathcal{U}'_\Xi(X, M) &\text{ iff } \varphi_s^{(X,M)} \text{ is a tautology} \\ &\text{ iff } \left(\varphi_s^{(\emptyset,M)}\right)^{(X,M)} \text{ is a tautology} \\ &\text{ iff } s \in \mathcal{U}'_{\Xi^M}(X, M) \end{aligned}$$

So in the complete lattice $(2^M, \subseteq)$, the operators $\mathcal{U}'_\Xi(\cdot, M)$ and $\mathcal{U}'_{\Xi^M}(\cdot, M)$ coincide. Therefore, their least fixpoints coincide. □

We close this section by illustrating some of the ultimate semantics using the example seen earlier. In the introduction, we already hinted at the fact that deciding whether a given statement s is contained in the lower or upper bound of the ultimate revision of a given pair (X, Y) can be regarded as checking whether the partially evaluated acceptance formula $\varphi_s^{(X,Y)}$ is irrefutable (lower bound) or satisfiable (upper bound), respectively. For illustration purposes, we now make use of this fact here.

Example 2.1 (Continued). The deliberation in D_{swim} quite clearly yields that we should go for a swim, since the ultimate grounded pair is given by

$$\bar{g} = (\{clouds, wind, swim\}, \{clouds, wind, swim\})$$

[8] So the reduct Ξ^M really is an ADF since all acceptance formulas mention only statements from M.

corresponding to the two-valued interpretation [11]

$$\{clouds \mapsto \mathbf{t}, wind \mapsto \mathbf{t}, rain \mapsto \mathbf{f}, hot \mapsto \mathbf{f}, swim \mapsto \mathbf{t}\}$$

In words, there are clouds and it is not hot, there will be wind and no rain, and we should go for a swim. Since the ultimate grounded interpretation is already two-valued (an exact pair), this interpretation is also the unique two-valued model of the ADF D_{swim} as well as its single ultimate complete and ultimate preferred interpretation. There are 16 ultimate admissible and 50 ultimate conflict-free interpretations, but it is more interesting to look at the ultimate naive interpretations:

$$\begin{aligned}
\bar{n}_1 &= (\{clouds, wind, swim\}, \{clouds, wind, swim\}) = \bar{g} \\
\bar{n}_2 &= (\{clouds, rain, swim\}, S) \\
\bar{n}_3 &= (\{clouds, rain\}, \{clouds, wind, rain\}) \\
\bar{n}_4 &= (\{clouds, wind\}, \{clouds, wind, rain\})
\end{aligned}$$

The first pair is the single two-valued model. In the second pair, intuitively, it rains, but we go for a swim nonetheless (it is undecided whether it is hot and so there is a slight chance that our swim is justified by it being hot). In the third pair there is rain, there might be wind, it is not hot, and we do not swim; in the fourth pair, it is hot and unclear whether there is rain, but we do not go for a swim. In order to illustrate more technically why the pair $\bar{n}_3$ (for example) is naive, that is, $\leq_i$-maximal conflict-free, we can have a look at the partially evaluated acceptance conditions:

$$\begin{aligned}
\varphi_{clouds}^{\bar{n}_3} &= \mathbf{t} \\
\varphi_{wind}^{\bar{n}_3} &= \mathbf{t} \\
\varphi_{rain}^{\bar{n}_3} &= (clouds \land \neg wind)^{(\{clouds, rain\}, \{clouds, wind, rain\})} = \mathbf{t} \land \neg wind \equiv \neg wind \\
\varphi_{hot}^{\bar{n}_3} &= \mathbf{f} \\
\varphi_{swim}^{\bar{n}_3} &= (\neg rain \lor hot)^{(\{clouds, rain\}, \{clouds, wind, rain\})} = \neg \mathbf{t} \lor \mathbf{f} \equiv \mathbf{f}
\end{aligned}$$

Setting *clouds* and *rain* to true is justified since their respective partially evaluated acceptance formulas are satisfiable. Symmetrically, setting *hot* and *swim* to false is justified since their partially evaluated acceptance formulas are refutable. Setting *wind* to undecided need not be justified at all. This shows that $\bar{n}_3$ is conflict-free. To show that it is also naive, we have to show that all pairs $\bar{n}'$ with $\bar{n}_3 <_i \bar{n}'$ are *not* conflict-free. The only two candidates are

$$\begin{aligned}
\bar{n}' &= (\{clouds, wind, rain\}, \{clouds, wind, rain\}) \\
\bar{n}'' &= (\{clouds, rain\}, \{clouds, rain\})
\end{aligned}$$

For $\bar{n}'$, we get $\varphi_{rain}^{\bar{n}'} = \mathbf{t} \land \neg \mathbf{t} \equiv \mathbf{f}$, thus in $\bar{n}'$ setting *rain* to true is not justified, since its partially evaluated acceptance formula is unsatisfiable. For $\bar{n}''$, setting *wind* to false is unjustified in general since its acceptance formula is a tautology.

2.3. Complexity theory

We assume familiarity with the complexity classes P, NP and coNP, as well as with polynomial reductions and hardness and completeness for these classes (see [37] for a comprehensive introduction to complexity theory). We also make use of the polynomial hierarchy, that can be defined (using oracle Turing machines) as follows: $\Sigma_0^P = \Pi_0^P = \Delta_0^P = \mathsf{P}$, $\Sigma_{i+1}^P = \mathsf{NP}^{\Sigma_i^P}$, $\Pi_{i+1}^P = \mathsf{coNP}^{\Sigma_i^P}$,

$\Delta^P_{i+1} = \mathsf{P}^{\Sigma^P_i}$ for $i \geq 0$. For complete problems of the polynomial hierarchy we use here mainly satisfiability of quantified Boolean formulas (QBFs). The problem $\text{QBF}_{i,Q}$-TRUTH denotes the problem of deciding satisfiability of a given closed QBF in prenex form, starting with quantifier $Q \in \{\exists, \forall\}$ and i quantifier alternations. For $i \geq 0$ it holds that $\text{QBF}_{i,\exists}$-TRUTH is Σ^P_i-complete and $\text{QBF}_{i,\forall}$-TRUTH is Π^P_i-complete.

As a somewhat non-standard polynomial hierarchy complexity class, we use D^P_i, a generalization of the complexity class DP to the polynomial hierarchy. A language is in DP iff it is the intersection of a language in NP and a language in coNP. Generally, a language is in D^P_i iff it is the intersection of a language in Σ^P_i and a language in Π^P_i. The canonical problem of $\mathsf{DP} = \text{D}^P_1$ is SAT-UNSAT, the problem to decide for a given pair (ψ_1, ψ_2) of propositional formulas whether ψ_1 is satisfiable and ψ_2 is unsatisfiable. Obviously, by definition $\Sigma^P_i, \Pi^P_i \subseteq \text{D}^P_i \subseteq \Delta^P_{i+1}$ for all $i \geq 0$.

3. Preparatory Considerations

This section sets the stage and provides several technical preparations that will simplify our complexity analysis that follows afterwards. We first introduce some notation to make formally precise what decision problems we will analyze (Section 3.1). We then briefly recapitulate the currently known complexity results for ADFs in Section 3.2. Next, in Section 3.3 we study the relationship between the approximate and ultimate operator, where it will turn out that the operators are quite similar, yet subtly different. Section 3.4 provides two quite general existence results. They guarantee that approximating operators on CPOs always possess preferred and naive pairs, which will have an impact on the problem of deciding the existence of non-trivial pairs for these semantics. Since several of our hardness results use similar reduction techniques, we introduce some of them in Section 3.5 and prove properties that we will later use in hardness proofs. In Section 3.6 we analyze the complexity of computing the two operators we consider in this paper. Since the semantics that we study are defined within the framework of approximation fixpoint theory, knowing the complexity of operator computation is a valuable guide for investigating the operator-based semantics. Finally, in Section 3.7 we give generic results on upper bounds for operator-based semantics that only make use of upper bounds for the respective operators.

3.1. Notation and decision problems

For a set S, let

- $(A^c, \leq_i)$ be the consistent CPO of S-subset pairs,
- $\mathcal{O}$ an approximating operator on $(A^c, \leq_i)$.

In the following we tacitly assume that from a given approximation operator $\mathcal{O}$ one can infer the context CPO and the underlying set S, unless noted otherwise.

Let $\mathcal{A}$ be the set of all approximation operators, such that each is defined on some consistent CPO of S-subset pairs for some set S. We define decision problems with two parameters. The first is a set of approximation operators $\mathcal{I} \subseteq \mathcal{A}$. In addition to $\mathcal{A}$ we are interested in this paper in the following sets of operators.

- $\mathcal{G} = \{\mathcal{G}_\Xi \mid \Xi \text{ is an ADF}\}$,
- $\mathcal{U} = \{\mathcal{U}_\Xi \mid \Xi \text{ is an ADF}\}$

That is, the sets contain approximate, respectively ultimate operators for each possible ADF. When restricted to *bipolar* ADFs we denote the corresponding sets with $\mathcal{BG} = \{\mathcal{G}_\Xi \mid \Xi \text{ is a BADF}\}$ and $\mathcal{BU} = \{\mathcal{U}_\Xi \mid \Xi \text{ is a BADF}\}$. Clearly we have $\mathcal{G}, \mathcal{U} \subseteq \mathcal{A}$ and thus also $\mathcal{BG}, \mathcal{BU} \subseteq \mathcal{A}$. The semantics is the second parameter of our decision problems. Let $\sigma \in \{\textit{cfi}, \textit{nai}, \textit{adm}, \textit{com}, \textit{grd}, \textit{pre}, \textit{2su}, \textit{2st}\}$ be a semantics among conflict-free, naive, admissible, complete, grounded, preferred, two-valued supported and two-valued stable semantics, respectively. We first consider the *verification* problem, which asks if for a given operator a given pair is a σ-pair, respectively a σ-model.

Problem: $\mathsf{Ver}_\sigma^{\mathcal{I}}$
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$ and a pair $(X, Y) \in A^c$.
Question: Is (X, Y) a σ-model/pair of $\mathcal{O}$?

For instance $\mathsf{Ver}_{adm}^{\mathcal{G}}$ asks whether for a given approximate operator $\mathcal{G}_\Xi$ and $(X, Y) \in A^c$, does it hold that $(X, Y) \leq_i \mathcal{G}_\Xi(X, Y)$? The next decision problem asks whether there *exists a non-trivial* σ-pair/model, that is, one that is different from $(\emptyset, S)$.

Problem: $\mathsf{Exists}_\sigma^{\mathcal{I}}$
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$.
Question: Does there exist a σ-model/pair (X, Y) of $\mathcal{O}$ such that $(X, Y) \neq (\emptyset, S)$?

The remaining two decision problems define query-based reasoning. The *credulous* acceptance problem asks whether an element $s \in S$ is in X of at least one σ-pair/model (X, Y) of a given operator, while *skeptical* acceptance asks if this is the case for all σ-pairs/models.

Problem: $\mathsf{Cred}_\sigma^{\mathcal{I}}$
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$ and $s \in S$.
Question: Does there exist a σ-model/pair (X, Y) of $\mathcal{O}$ such that $s \in X$?

Problem: $\mathsf{Skept}_\sigma^{\mathcal{I}}$
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$ and $s \in S$.
Question: Does it hold that for all σ-models/pairs (X, Y) of $\mathcal{O}$ we have $s \in X$?

We now introduce auxiliary decision problems, which aid us in showing the computational complexity of revising the lower and upper bounds for a given approximation operator and pair. The first asks whether an element is in the revised lower bound (respectively upper bound) for a given pair.

Problem: $\mathsf{Elem}^{\mathcal{I}'}$ (resp. $\mathsf{Elem}^{\mathcal{I}''}$)
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$, a pair $(X, Y) \in A^c$ and $s \in S$.
Question: Does it hold that $s \in \mathcal{O}'(X, Y)$? (resp. $s \in \mathcal{O}''(X, Y)$)

Let $\circ \in \{\subseteq, \supseteq\}$. The next decision problem considers all combinations of asking whether for a given pair and approximation operator the given set is a subset/superset of the revised lower/upper bound.

Problem: $\mathsf{RevBound}_\circ^{\mathcal{I}'}$
Instance: An approximation operator $\mathcal{O} \in \mathcal{I}$, a pair $(X, Y) \in A^c$ and a set $B \subseteq S$.
Question: if $\circ = \subseteq$: Is $B \subseteq \mathcal{O}'(X, Y)$?
if $\circ = \supseteq$: Is $\mathcal{O}'(X, Y) \subseteq B$?

Similarly, $\mathsf{RevBound}_\circ^{\mathcal{I}''}$ denotes the variant for the revision of the upper bound ($\mathcal{O}''$). For instance $\mathsf{RevBound}_\supseteq^{\mathcal{I}''}$ denotes the problem of checking whether for an approximation operator $\mathcal{O} \in \mathcal{I}$, $B \subseteq S$ and a given pair $(X, Y) \in A^c$ we have $\mathcal{O}''(X, Y) \subseteq B$, that is, if the set is a superset of the revised upper bound (indicated by $\cdot''$).

3.2. Existing results

We briefly survey – to the best of our knowledge – all existing complexity results for abstract dialectical frameworks. For general ADFs and the ultimate family of semantics, Brewka et al. [11] have shown the following:

- $\mathsf{Ver}^{\mathcal{U}}_{2su}$ is in P, $\mathsf{Exists}^{\mathcal{U}}_{2su}$ is NP-complete, (Proposition 5)
- $\mathsf{Ver}^{\mathcal{U}}_{adm}$ is coNP-complete, (Proposition 10)
- $\mathsf{Ver}^{\mathcal{U}}_{grd}$ and $\mathsf{Ver}^{\mathcal{U}}_{com}$ are DP-complete, (Theorem 6, Cor. 7)
- $\mathsf{Ver}^{\mathcal{U}}_{2st}$ is in DP, (Proposition 8)
- $\mathsf{Exists}^{\mathcal{U}}_{2st}$ is Σ^P_2-complete. (Theorem 9)

For bipolar ADFs, Brewka and Woltran [9] showed that $\mathsf{Ver}^{\mathcal{BU}}_{grd}$ is in P (Proposition 15). So particularly for BADFs, this paper will greatly illuminate the complexity landscape.

3.3. Relationship between the operators

Since $\mathcal{U}_\Xi$ is the ultimate approximation of G_Ξ for an ADF Ξ it is clear that for any $X \subseteq Y \subseteq S$ we have $\mathcal{G}_\Xi(X,Y) \leq_i \mathcal{U}_\Xi(X,Y)$. In other words, the ultimate revision operator produces new bounds that are at least as tight as those of the approximate operator. More explicitly, the ultimate new lower bound always contains the approximate new lower bound: $\mathcal{G}'_\Xi(X,Y) \subseteq \mathcal{U}'_\Xi(X,Y)$; conversely, the ultimate new upper bound is contained in the approximate new upper bound: $\mathcal{U}''_\Xi(X,Y) \subseteq \mathcal{G}''_\Xi(X,Y)$. Somewhat surprisingly, it turns out that the revision operators for the upper bound coincide.

Lemma 3.1. *Let $\Xi = (S, L, C)$ be an ADF and $X \subseteq Y \subseteq S$.*

$$\mathcal{G}''_\Xi(X,Y) = \mathcal{U}''_\Xi(X,Y)$$

Proof. Let $s \in S$. We will use that for all $B, X, P \subseteq S$, we find $(P \setminus B) \cap X = \emptyset$ iff $P \cap X \subseteq B$. Now

$$\begin{aligned} s \in \mathcal{G}''_\Xi(X,Y) \text{ iff } & \exists B : B \subseteq par(s) \cap Y \text{ and } C_s(B) = \mathbf{t} \text{ and } (par(s) \setminus B) \cap X = \emptyset \\ \text{iff } & \exists B : par(s) \cap X \subseteq B \subseteq par(s) \cap Y \text{ and } C_s(B) = \mathbf{t} \\ \text{iff } & \exists Z : X \subseteq Z \subseteq Y \text{ and } C_s(Z \cap par(s)) = \mathbf{t} \\ \text{iff } & s \in \mathcal{U}''_\Xi(X,Y) \end{aligned}$$

□

The operators for computing a new lower bound are demonstrably different, since we can find Ξ and (X,Y) with $\mathcal{U}'_\Xi(X,Y) \not\subseteq \mathcal{G}'_\Xi(X,Y)$, as the following ADF shows.

Example 3.1. Consider the ADF $D = (\{a\}, \{(a,a)\}, \{\varphi_a\})$ with one self-dependent statement a that has acceptance formula $\varphi_a = a \vee \neg a$. In Figure 1, we show the relevant CPO and the behavior of approximate and ultimate operators: we see that $\mathcal{G}_D(\emptyset, \{a\}) <_i \mathcal{U}_D(\emptyset, \{a\})$, which shows that in some cases the ultimate operator is strictly more precise.

So in a sense the approximate operator cannot see beyond the case distinction $a \vee \neg a$. As we will see shortly, this difference really amounts to the capability of tautology checking.

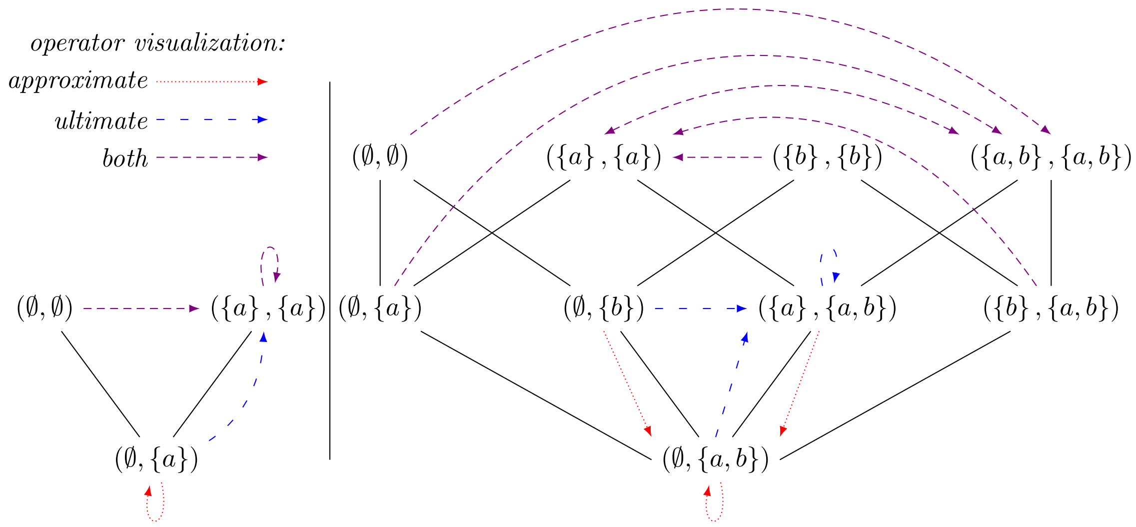


Figure 1: Hasse diagrams of consistent CPOs for the ADFs from Example 3.1 (left) and Example 3.2 (right). Solid lines represent the information ordering $\leq_i$. Directed arrows express how revision operators map pairs to other pairs. For pairs where the revisions coincide, the arrows are densely dashed and violet. When the operators revise a pair differently, we use a dotted red arrow for the ultimate and a loosely dashed blue arrow for the approximate operator. Exact (two-valued) pairs are the $\leq_i$-maximal elements. For those pairs, (and any ADF Ξ) it is clear that the operators $\mathcal{U}_\Xi$ and $\mathcal{G}_\Xi$ coincide since they approximate the same two-valued operator G_Ξ. In Example 3.1 on the left, we can see that the ultimate operator maps all pairs to its only fixpoint $(\{a\}, \{a\})$ where a is true. The approximate operator has an additional fixpoint, $(\emptyset, \{a\})$, where a is unknown. In Example 3.2 on the right, the major difference between the operators is whether statement a can be derived given that b has truth value unknown. This is the case for the ultimate, but not for the approximate operator. Since there is no fixpoint in the upper row (showing the two-valued operator G_E), the ADF E does not have a two-valued model. Each of the revision operators has however exactly one three-valued fixpoint, which thus constitutes the respective grounded, preferred and complete semantics.

Example 3.2. ADF $E = (\{a, b\}, \{(b, a), (b, b)\}, \{\varphi_a, \varphi_b\})$ has acceptance formulas $\varphi_a = b \vee \neg b$ and $\varphi_b = \neg b$. So b is self-attacking and the link from b to a is redundant. In Figure 1, we show the relevant CPO and the behavior of the operators $\mathcal{U}_E$ and $\mathcal{G}_E$ on this CPO.

The examples show that the approximate and ultimate families of semantics really are different, save for one straightforward inclusion relation in case of admissible.

Corollary 3.2. *For any ADF Ξ it holds that an admissible pair of $\mathcal{G}_\Xi$ is an admissible pair of $\mathcal{U}_\Xi$. Let $\sigma \in \{com, grd, pre\}$. There exist ADFs Ξ_1, Ξ_2, Ξ_3 such that:*

1. *there is an admissible pair of $\mathcal{U}_{\Xi_1}$ that is not an admissible pair of $\mathcal{G}_{\Xi_1}$;*
2. *there is a σ-pair of $\mathcal{U}_{\Xi_2}$ that is not a σ-pair of $\mathcal{G}_{\Xi_2}$; and*
3. *there is a σ-pair of $\mathcal{G}_{\Xi_3}$ that is not a σ-pair of $\mathcal{U}_{\Xi_3}$*

Proof. To show that an approximate admissible pair is always an ultimate admissible pair it suffices to consider the fact that $\mathcal{G}_\Xi \leq_i \mathcal{U}_\Xi$. For the remaining claims, we use $\Xi_1 = \Xi_2 = \Xi_3 = E$ from Example 3.2 as a witness:

1. *In Example 3.2, $(\{a\}, \{a,b\})$ is ultimate admissible but not approximate admissible.*

2 & 3. *In Example 3.2, we have: (1) approximate grounded, preferred and complete semantics coincide; (2) ultimate grounded, preferred and complete semantics coincide; (3) approximate grounded and ultimate grounded semantics are different with no subset relation either way.* □

3.4. Existence results

We next present two general theorems that guarantee the existence of certain pairs for approximating operators on CPOs. By CPOs here we do not only refer to S-subset CPOs $(A^c, \leq_i)$, but in fact to arbitrary CPOs $(L^c, \leq_i)$ containing consistent pairs of elements of a complete lattice $(L, \sqsubseteq)$. Both results make use of the axiom of choice – the second one directly, and the first one in the form of Zorn's lemma. The first result says that for each admissible pair there is a preferred pair containing at least as much information. This significantly generalizes a result by Dung ([24, Theorem 11]) to general operators.

Theorem 3.3. *Let $(L, \sqsubseteq)$ be a complete lattice and $\mathcal{O}$ an approximating operator on the CPO $(L^c, \leq_i)$. For each admissible pair $\bar{a} \in L^c$, there exists a preferred pair $\bar{p} \in L^c$ with $\bar{a} \leq_i \bar{p}$.*

Proof. *Let $\bar{a} \in L^c$ with $\bar{a} \leq_i \mathcal{O}(\bar{a})$. Define the set of all $\mathcal{O}$-admissible pairs that contain at least as much information as $\bar{a}$,*

$$C = \{\bar{c} \mid \bar{a} \leq_i \bar{c} \text{ and } \bar{c} \leq_i \mathcal{O}(\bar{c})\}$$

We show that $(C, \leq_i)$ is a CPO. Clearly $\bar{a} \in C$ is the least element of the poset $(C, \leq_i)$. Now let $D \subseteq C$ be directed and $\bar{e} = \bigsqcup_{L^c} D$ be its least upper bound in L^c. We show $\bar{e} \in C$, that is, $\bar{a} \leq_i \bar{e}$ and $\bar{e} \leq_i \mathcal{O}(\bar{e})$. Since D is directed, it is non-empty, so there is some $\bar{z} \in D$, whence $\bar{a} \leq_i \bar{z} \leq_i \bar{e}$. Now for each $\bar{z} \in D$, we have $\bar{z} \leq_i \bar{e}$ since $\bar{e}$ is an upper bound of D. Since $\mathcal{O}$ is $\leq_i$-monotone, we have $\mathcal{O}(\bar{z}) \leq_i \mathcal{O}(\bar{e})$. Since $\bar{z} \in D \subseteq C$, by definition $\bar{z} \leq_i \mathcal{O}(\bar{z})$. In combination, $\bar{z} \leq_i \mathcal{O}(\bar{z}) \leq_i \mathcal{O}(\bar{e})$. Thus $\mathcal{O}(\bar{e})$ is an upper bound of D. Since $\bar{e}$ is the least upper bound of D, we have $\bar{e} \leq_i \mathcal{O}(\bar{e})$.

Thus $(C, \leq_i)$ is a CPO and therefore each ascending chain has an upper bound in C. By Zorn's lemma, C has a $\leq_i$-maximal element $\bar{p} \in C$, which by $\bar{a} \leq_i \bar{p}$ is the desired preferred pair. □

Theorem 3.3 directly leads to the next result, which considerably simplifies the complexity analysis of deciding the existence of non-trivial pairs for admissibility-based semantics.

Lemma 3.4. *Let $(L, \sqsubseteq)$ be a complete lattice and $\mathcal{O}$ an approximating operator on the CPO $(L^c, \leq_i)$. The following are equivalent:*

1. *$\mathcal{O}$ has a non-trivial admissible pair.*
2. *$\mathcal{O}$ has a non-trivial preferred pair.*
3. *$\mathcal{O}$ has a non-trivial complete pair.*

Proof. "(1) ⇒ (2)": *Let $(\bot_L, \top_L) <_i (x, y) \leq_i \mathcal{O}(x, y)$. By Theorem 3.3, there is a preferred pair $(p, q) \in L^c$ for which $(\bot_L, \top_L) <_i (x, y) \leq_i (p, q)$.*

"(2) ⇒ (3)": *By [40, Theorem 3.10], every preferred pair is complete.*

"(3) ⇒ (1)": *Any complete pair is admissible (Table 1).* □

This directly shows the equivalence of the respective decision problems, that is, it holds that $\mathsf{Exists}^{\mathcal{A}}_{adm} = \mathsf{Exists}^{\mathcal{A}}_{pre} = \mathsf{Exists}^{\mathcal{A}}_{com}$. Recall that $\mathcal{A}$ contains all approximation operators defined on some consistent CPO of S-subset pairs for some set S. Regarding decision problems for querying, skeptical reasoning with respect to admissibility is trivial, that is, $(\emptyset, S)$ is always an admissible pair in any ADF. Furthermore, credulous reasoning with respect to admissible, complete and preferred semantics coincides.

Lemma 3.5. *Let $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. It holds that $\mathsf{Cred}^{\mathcal{I}}_{adm} = \mathsf{Cred}^{\mathcal{I}}_{com} = \mathsf{Cred}^{\mathcal{I}}_{pre}$.*

Proof. *Let Ξ be an ADF, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$ and $s \in S$. Assume (X, Y) with $s \in X$ is admissible w.r.t. $\mathcal{O}$, then there exists a (X', Y') with $(X, Y) \leq_i (X', Y')$ which is preferred with respect to $\mathcal{O}$ and where $s \in X'$ by Theorem 3.3. Since any preferred pair is also complete and any complete pair is also admissible the claim follows.* □

For semantics based on conflict-freeness, an existence result similar to Theorem 3.3 holds. The proof is inspired by the proof of [7, Theorem 1] (see also [17, Theorem 8.23], in particular for the concept of "roofs"), and sufficiently complicated. The major part of the proof is concerned with showing that there is a chain of conflict-free elements that starts with the given conflict-free element, and that this chain is itself a CPO. Again, the result is not restricted to subset-CPOs.

Theorem 3.6. *Let $(L, \sqsubseteq)$ be a complete lattice and $\mathcal{O}$ an approximating operator on the CPO $(L^c, \leq_i)$. For each conflict-free pair $\bar{c} \in L^c$, there exists a naive pair $\bar{n} \in L^c$ with $\bar{c} \leq_i \bar{n}$.*

Proof. *Let $\bar{c} \in L^c$ be conflict-free. Define the set*

$$D = \{\bar{a} \in L^c \mid \bar{c} \leq_i \bar{a}\}$$

Clearly $(D, \leq_i)$ is a CPO with least element $\bar{c}$. (Its least upper bound is given by $\sqcup_D = \sqcup_{L^c}$.) For any conflict-free pair $\bar{a} \in D$ that is not naive, by definition there exists a conflict-free pair $\bar{a}' \in D$ such that $\bar{a} <_i \bar{a}'$. Thus by the axiom of choice, there exists a function $f : D \to D$ with

$$\bar{a} \mapsto \begin{cases} \bar{a}' & \text{if } \bar{a} \text{ is conflict-free, but not naive} \\ \bar{a} & \text{otherwise} \end{cases}$$

Clearly f is increasing, that is, for all $\bar{a} \in D$ we have $\bar{a} \leq_i f(\bar{a})$. Furthermore, $f(\bar{a})$ is conflict-free iff $\bar{a}$ is conflict-free. Thus a conflict-free pair $\bar{a}$ is a fixpoint of f iff $\bar{a}$ is naive. We proceed to show that such a fixpoint exists.

We look at the smallest f-closed sub-CPO of $(D, \leq_i)$, that is, the smallest set $F \subseteq D$ such that $f(F) \subseteq F$ and $(F, \leq_i)$ is a CPO. Clearly its least element is $\bot_F = \bar{c}$, the least element of D.

We call an element $\bar{u} \in F$ a roof iff for all $\bar{v} \in F$ with $\bar{v} <_i \bar{u}$ we have $f(\bar{v}) \leq_i \bar{u}$. For each pair $\bar{u} \in F$, we show that if $\bar{u}$ is a roof, then the set

$$Z_{\bar{u}} = \{\bar{v} \in F \mid \bar{v} \leq_i \bar{u} \text{ or } f(\bar{u}) \leq_i \bar{v}\}$$

is an f-closed sub-CPO of $(F, \leq_i)$. So let $\bar{u} \in F$ be a roof and consider $Z_{\bar{u}}$. We have to show that $f(Z_{\bar{u}}) \subseteq Z_{\bar{u}}$ and $(Z_{\bar{u}}, \leq_i)$ is a CPO.

$f(Z_{\bar{u}}) \subseteq Z_{\bar{u}}$: *Let $\bar{v} \in Z_{\bar{u}}$. Then $\bar{v} \leq_i \bar{u}$ or $f(\bar{u}) \leq_i \bar{v}$. We have to show $f(\bar{v}) \in Z_{\bar{u}}$, that is, $f(\bar{v}) \leq_i \bar{u}$ or $f(\bar{u}) \leq_i f(\bar{v})$. If $f(\bar{u}) \leq_i \bar{v}$, then since f is increasing we get $f(\bar{u}) \leq_i \bar{v} \leq_i f(\bar{v})$. If $\bar{v} <_i \bar{u}$, then since $\bar{u}$ is a roof we get $f(\bar{v}) \leq_i \bar{u}$. If $\bar{v} = \bar{u}$ then $f(\bar{u}) \leq_i f(\bar{v})$ is clear.*

$(Z_{\bar{u}}, \leq_i)$ is a CPO: $\perp_F \in Z_{\bar{u}}$ *is the least element of the poset* $(Z_{\bar{u}}, \leq_i)$. *Let* $E \subseteq Z_{\bar{u}}$ *be directed and* $\bar{e} = \bigsqcup_F E$ *be its least upper bound in* $(F, \leq_i)$. *We have to show* $\bar{e} \in Z_{\bar{u}}$, *that is,* $\bar{e} \leq_i \bar{u}$ *or* $f(\bar{u}) \leq_i \bar{e}$. *By assumption,*

$$Z_{\bar{u}} = Z^l_{\bar{u}} \cup Z^r_{\bar{u}} \text{ with } Z^l_{\bar{u}} = \{\bar{v} \in F \mid \bar{v} \leq_i \bar{u}\} \text{ and } Z^r_{\bar{u}} = \{\bar{v} \in F \mid f(\bar{u}) \leq_i \bar{v}\}$$

Define $E^l = E \cap Z^l_{\bar{u}}$ *and* $E^r = E \cap Z^r_{\bar{u}}$. *Clearly* $\bar{u}$ *is an upper bound of* E^l *and* $f(\bar{u})$ *is a lower bound of* E^r; *moreover* $\bar{e}$ *is an upper bound of* E^r. *Thus if* $E^r \neq \emptyset$ *then* $f(\bar{u}) \leq_i \bar{e}$ *and we are done. Otherwise* $E^r = \emptyset$, *then* $E = E^l$ *and* $\bar{u}$ *is an upper bound of* E. *Since* $\bar{e}$ *is the least upper bound of* E, *we get* $\bar{e} \leq_i \bar{u}$.

Thus if $\bar{u} \in F$ *is a roof then* $(Z_{\bar{u}}, \leq_i)$ *with* $Z_{\bar{u}} \subseteq F$ *is an* f*-closed sub-CPO of* $(D, \leq_i)$. *Since* $(F, \leq_i)$ *is the least* f*-closed sub-CPO of* $(D, \leq_i)$, *we get* $F \subseteq Z_{\bar{u}}$ *and thus* $Z_{\bar{u}} = F$ *for each roof* $\bar{u} \in F$. *Now we show that each pair* $\bar{u} \in F$ *is a roof. Define the set* $U = \{\bar{u} \in F \mid \bar{u} \text{ is a roof}\}$. *We show that* $(U, \leq_i)$ *is an* f*-closed sub-CPO of* $(F, \leq_i)$.

$f(U) \subseteq U$: *Let* $\bar{u} \in U$. *Then for all* $\bar{v} \in F$ *with* $\bar{v} <_i \bar{u}$ *we have* $f(\bar{v}) \leq_i \bar{u}$. *We have to show* $f(\bar{u}) \in U$, *that is, for all* $\bar{v} \in F$ *with* $\bar{v} <_i f(\bar{u})$ *we have* $f(\bar{v}) \leq_i f(\bar{u})$.

Let $\bar{v} \in F$ *with* $\bar{v} <_i f(\bar{u})$. *Since* $\bar{v} \in F = Z_{\bar{u}}$, *we find that* $\bar{v} \leq_i \bar{u}$ *or* $f(\bar{u}) \leq_i \bar{v}$. *Note that* $f(\bar{u}) \leq_i \bar{v}$ *is impossible by presumption. If* $\bar{v} <_i \bar{u}$ *then we have* $f(\bar{v}) \leq_i \bar{u} \leq_i f(\bar{u})$ *by presumption. If* $\bar{v} = \bar{u}$ *then* $f(\bar{v}) \leq_i f(\bar{u})$ *is clear.*

$(U, \leq_i)$ is a CPO: $\perp_F$ *is trivially a roof, whence* $\perp_F \in U$. *Now let* $W \subseteq U$ *be directed and let* $\bar{w} = \bigsqcup_F W$ *be the least upper bound of* W *in* F. *We show* $\bar{w} \in U$, *that is, for all* $\bar{v} \in F$ *with* $\bar{v} <_i \bar{w}$ *we have* $f(\bar{v}) \leq_i \bar{w}$.

Let $\bar{v} \in F$ *with* $\bar{v} <_i \bar{w}$. *If for all* $\bar{z} \in W$ *we had* $\bar{z} \leq_i \bar{v}$, *then* $\bar{v}$ *would be an upper bound of* W, *whence* $\bar{w} \leq_i \bar{v}$ *contrary to assumption. Thus there is a* $\bar{z} \in W$ *with* $\bar{z} \not\leq_i \bar{v}$. *Now* $\bar{z} \in W \subseteq U$ *is a roof, and we have* $\bar{v} \in F = Z_{\bar{z}}$, *that is,* $\bar{v} \leq_i \bar{z}$ *or* $f(\bar{z}) \leq_i \bar{v}$. *Due to* $\bar{z} \leq_i f(\bar{z})$ *and* $\bar{z} \not\leq_i \bar{v}$ *we get* $\bar{v} \leq_i \bar{z}$; *additionally,* $\bar{z} \leq_i \bar{w}$ *since* $\bar{w}$ *is an upper bound of* W. *Now if* $\bar{v} = \bar{z}$ *then* $\bar{v}$ *is a roof and* $\bar{w} \leq_i \bar{v}$ *or* $f(\bar{v}) \leq_i \bar{w}$, *where the first is impossible by presumption. Finally, if* $\bar{v} <_i \bar{z}$ *then* $\bar{z}$ *being a roof implies that* $f(\bar{v}) \leq_i \bar{z} \leq_i \bar{w}$.

Thus $(U, \leq_i)$ *with* $U \subseteq F$ *is an* f*-closed sub-CPO of* $(D, \leq_i)$. *Since* $(F, \leq_i)$ *is the least* f*-closed sub-CPO of* $(D, \leq_i)$, *we have* $F \subseteq U$, *that is,* $F = U$.

Now we show that F *is a chain, that is, for all* $\bar{u}, \bar{v} \in F$ *we find* $\bar{u} \leq_i \bar{v}$ *or* $\bar{v} \leq_i \bar{u}$: *since* $\bar{u}$ *is a roof,* $\bar{v} \in F = Z_{\bar{u}}$ *whence* $\bar{v} \leq_i \bar{u}$ *or* $\bar{u} \leq_i f(\bar{u}) \leq_i \bar{v}$. *Now* F *is a CPO and a chain, it therefore has a least upper bound in* F, *that is, a greatest element* $\top_F = \bigsqcup_F F$. *Since* f *is increasing, we have* $\top_F \leq_i f(\top_F)$; *since* F *is* f*-closed,* $f(\top_F) \in F$; *since* $\top_F$ *is the greatest element of* F, *we find* $f(\top_F) \leq_i \top_F$. *Thus* $\top_F$ *is a fixpoint of* f. *It remains to show that* $\top_F$ *is conflict-free. In fact, all elements of* F *are conflict-free: assume there were a* $\bar{v} \in F$ *that was not conflict-free, then* $f^{-1}(\bar{v}) = \{\bar{v}\}$ *by definition and* $(F \setminus \{\bar{v}\}, \leq_i)$ *would be an* f*-closed proper sub-CPO of* F, *contradiction. Consequently,* $\bar{n} = \top_F$ *with* $\bar{c} = \perp_F \leq_i \top_F = \bar{n}$ *is our desired naive pair.* □

From the last part of the proof it might seem that the desired naive pair is uniquely determined. This is however not the case – the application of the axiom of choice in the beginning gives us an arbitrary chain of conflict-free pairs, there might be many more in $(L^c, \leq_i)$.

As in the case of admissible-based semantics, the existence of non-trivial naive pairs is then equivalent to the existence of non-trivial conflict-free pairs.

Lemma 3.7. *Let* $(L, \sqsubseteq)$ *be a complete lattice and* $\mathcal{O}$ *an approximating operator on* $(L^c, \leq_i)$. *The following are equivalent:*

1. $\mathcal{O}$ *has a non-trivial conflict-free pair.*
2. $\mathcal{O}$ *has a non-trivial naive pair.*

Proof. "(1) ⇒ (2)": *Let (x, y) be non-trivial and conflict-free, that is, in particular let $(\bot_L, \top_L) <_i (x, y)$. By Theorem 3.6, there exists a naive pair $(p, q) \in L^c$ with $(\bot_L, \top_L) <_i (x, y) \leq_i (p, q)$.*

"(2) ⇒ (1)": *Any naive pair is conflict-free (Table 1).* □

Again, this directly shows the equivalence of the respective decision problems, that is, it holds that $\mathsf{Exists}^{\mathcal{A}}_{cfi} = \mathsf{Exists}^{\mathcal{A}}_{nai}$.

We finally prove a useful technical result that gives some insight into the structure of sets of conflict-free interpretations, namely, that such sets are downward-closed with respect to the CPO ordering. Notably, again, this result holds for arbitrary approximating operators.

Lemma 3.8. *Let $(L, \sqsubseteq)$ be a complete lattice and $\mathcal{O}$ an approximating operator on the CPO $(L^c, \leq_i)$. If $(x, y) \in L^c$ is conflict-free for $\mathcal{O}$, then so is any $(u, v) \leq_i (x, y)$.*

Proof. *Let $(x, y) \in L^c$ be conflict-free for $\mathcal{O}$ and $(u, v) \leq_i (x, y)$. First observe that this means $x \sqsubseteq \mathcal{O}''(x, y)$, $\mathcal{O}'(x, y) \sqsubseteq y$ and $u \sqsubseteq x \sqsubseteq y \sqsubseteq v$. Now since $\mathcal{O}$ is approximating, it is in particular $\leq_i$-monotone and thus $\mathcal{O}(u, v) \leq_i \mathcal{O}(x, y)$, that is,*

$$\mathcal{O}'(u, v) \sqsubseteq \mathcal{O}'(x, y) \quad \text{and} \quad \mathcal{O}''(x, y) \sqsubseteq \mathcal{O}''(u, v)$$

Combining all of the above, it follows that

$$\begin{aligned} u \sqsubseteq x \sqsubseteq \mathcal{O}''(x, y) \sqsubseteq \mathcal{O}''(u, v) \\ \mathcal{O}'(u, v) \sqsubseteq \mathcal{O}'(x, y) \sqsubseteq y \sqsubseteq v \end{aligned}$$

whence (u, v) is conflict-free for $\mathcal{O}$. □

3.5. Reductions and Encoding Techniques

In the sequel we apply several reductions for showing complexity-analytic results. In this section we present recurring reductions as well as certain "encoding schemes" that will prove useful. First we show that statements with self-conflicting acceptance conditions are always undecided in all conflict-free pairs.

Lemma 3.9. *Let $\Xi = (S, L, C)$ be an ADF, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$, $s \in S$ and $\varphi_s = \neg s$. For every conflict-free pair (X, Y) of $\mathcal{O}$ it holds that $s \in (Y \setminus X)$.*

Proof. *Assume (X, Y) is conflict-free for $\mathcal{O}$. By definition of conflict-free pairs we have $X \subseteq \mathcal{O}''(X, Y)$ and $\mathcal{O}'(X, Y) \subseteq Y$. We prove that s cannot be true or false in this pair. Suppose $s \in X$. We now show that this implies $s \notin \mathcal{O}''(X, Y)$. We have $X \not\models \varphi_s = \neg s$. It follows that for any Z with $X \subseteq Z \subseteq Y$ we have $Z \not\models \varphi_s$ and thus $s \notin \mathcal{O}''(X, Y)$ (approximate and ultimate operators coincide on the revised upper bound; see also Lemma 3.1). This implies that $X \not\subseteq \mathcal{O}''(X, Y)$ which is a contradiction to the definition of conflict-freeness.*

Suppose now that $s \in (S \setminus Y)$. We show that this implies $s \in \mathcal{G}'_\Xi(X, Y)$ and thus also $s \in \mathcal{U}'_\Xi(X, Y)$ (the ultimate operator is at least as precise). Using the definition of the approximate operator, we have $\emptyset \subseteq X$, $par(s) = \{s\}$, $(\{s\} \setminus \emptyset) \cap Y = \emptyset$, and $\emptyset \models \varphi_s$, and thus it follows that $s \in \mathcal{G}'_\Xi(X, Y)$. This implies that $\mathcal{O}'(X, Y) \not\subseteq Y$ which is again a contradiction to conflict-freeness. □

Self-attacking conditions can also be used to express integrity constraints in the following sense. If $\varphi_s = \neg s \wedge \phi$ for some formula ϕ then s is never assigned the value true in a conflict-free pair. Depending on the formula ϕ there might be cases under which s is assigned false or undecided, but we can exclude one truth value. We formalize this notion in the following Lemma. Note that the first item in the Lemma refers to both operators, while the second item refers only to the ultimate operator.

Lemma 3.10. *Let $\Xi = (S, L, C)$ be an ADF, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$, $s \in S$ and ϕ a formula over S.*

- *If $\varphi_s = \neg s \wedge \phi$ then there is no conflict-free pair (X, Y) of $\mathcal{O}$ such that $s \in X$.*
- *If $\varphi_s = \neg s \vee \phi$ then there is no conflict-free pair (X, Y) of $\mathcal{U}_\Xi$ such that $s \in (S \setminus Y)$.*

Proof. Suppose the contrary of the first item, i.e. that there is a conflict-free pair (X, Y) of $\mathcal{O}$ such that $s \in X$. By definition it holds that $X \subseteq \mathcal{O}''(X, Y)$. Since for all Z with $X \subseteq Z \subseteq Y$ it holds that $s \in Z$, it follows that $Z \not\models \varphi_s$ and thus $s \notin \mathcal{U}''_\Xi(X, Y)$ and, due to Lemma 3.1, also $s \notin \mathcal{G}''_\Xi(X, Y)$, which is a contradiction.

Suppose the contrary of the second item, i.e. that there is a conflict-free pair (X, Y) of $\mathcal{U}_\Xi$ such that $s \in (S \setminus Y)$. Then for all Z with $X \subseteq Z \subseteq Y$ it holds that $Z \models \varphi_s$. This implies that $s \in \mathcal{U}'_\Xi(X, Y)$ and therefore $\mathcal{U}'_\Xi(X, Y) \not\subseteq Y$, which is a contradiction. □

The next technique is similar to the previous one and makes sure that if a certain statement s is undecided in an admissible pair we can infer that a particular set $\{p, \text{-}p\}$ of statements must also be undecided. In this way undecidedness is "propagated" if we cannot assign a different value than undecided to the statement.

Lemma 3.11. *Let $\Xi = (S, L, C)$ be an ADF, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$, $s, p, \text{-}p \in S$ with*

- *$\varphi_p = \neg s \wedge \neg\text{-}p$ and*
- *$\varphi_{\text{-}p} = \neg s \wedge \neg p$.*

If (X, Y) is an admissible pair for $\mathcal{O}$ and $s \in (Y \setminus X)$, then $p, \text{-}p \in (Y \setminus X)$.

Proof. Assume (X, Y) with $s \in (Y \setminus X)$ is admissible for $\mathcal{O}$. We begin with the case for $\mathcal{O} = \mathcal{U}_\Xi$. It holds that $X \subseteq X \cup \{s\} \subseteq Y$, $X \cup \{s\} \not\models \varphi_p$, and $X \cup \{s\} \not\models \varphi_{\text{-}p}$. Thus $p, \text{-}p \notin \mathcal{U}'_\Xi(X, Y)$. Since $X \subseteq \mathcal{U}'_\Xi(X, Y)$ we can infer that $p, \text{-}p \notin X$.

Suppose $p \in (S \setminus Y)$. By definition of admissibility we have $p \notin \mathcal{U}''_\Xi(X, Y)$. Then for all Z with $X \subseteq Z \subseteq Y$ we have $Z \not\models \varphi_p$. It holds that $Z \not\models \varphi_p$ iff $s \in Z$ or $\text{-}p \in Z$. Consider the case $Z = X$. Since $s \notin X$ (by assumption), for $X \not\models \varphi_p$ to hold it must be the case that $\text{-}p \in X$. This is a contradiction, as shown above ($\text{-}p \notin X$). Therefore $p \notin (S \setminus Y)$ and it follows that $p \in (Y \setminus X)$. The proof that also $\text{-}p \in (Y \setminus X)$ holds is analogous.

For the approximate operator just observe that if $p, \text{-}p \notin \mathcal{U}'_\Xi(X, Y)$, then this implies that $p, \text{-}p \notin \mathcal{G}'_\Xi(X, Y)$, since $\mathcal{G}_\Xi \leq_i \mathcal{U}_\Xi$ (the ultimate operator is more precise). The remaining proof is analogous to the one for the ultimate operator, since the revisions of the upper bound coincide for both operators (Lemma 3.1). □

Self-supporting statements can be used to enforce the existence of preferred pairs, which assign true or false to this statement. This is in particular useful to "generate" preferred pairs for each two-valued valuation on a set of variables.

Lemma 3.12. *Let $\Xi = (S, L, C)$ be an ADF, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$, $s \in S$ and $\varphi_s = s$. It holds that*

- *there exist two preferred pairs (X, Y) and (X', Y') of $\mathcal{O}$ such that $s \in X$ and $s \in (S \setminus Y')$,*

- *for all preferred pairs (X, Y) of $\mathcal{O}$ it is the case that $s \notin (Y \setminus X)$.*

Proof. *Consider an arbitrary admissible pair (X, Y) for $\mathcal{O}$ with $s \in (Y \setminus X)$. Such a pair exists, since $(\emptyset, S)$ is a trivial admissible pair for $\mathcal{O}$. We have $(X, Y) \leq_i \mathcal{O}(X, Y)$ by admissibility. Since both pairs $(X \cup \{s\}, Y)$ and $(X, Y \setminus \{s\})$ are strictly more informative than (X, Y) it holds that $\mathcal{O}(X, Y) \leq_i \mathcal{O}(X \cup \{s\}, Y)$ and $\mathcal{O}(X, Y) \leq_i \mathcal{O}(X, Y \setminus \{s\})$ ($\mathcal{O}$ is $\leq_i$-monotone). We now show that both such pairs are admissible for $\mathcal{O}$. Consider the first pair $(X \cup \{s\}, Y)$. We know that $(X, Y) \leq_i \mathcal{O}(X \cup \{s\}, Y)$ ($\leq_i$ is transitive) and thus can infer both $X \subseteq \mathcal{O}'(X \cup \{s\}, Y)$ and $\mathcal{O}''(X \cup \{s\}, Y) \subseteq Y$. For showing that $(X \cup \{s\}, Y)$ is admissible, we only need to show that $X \cup \{s\} \subseteq \mathcal{O}'(X \cup \{s\}, Y)$ holds. To see that $s \in \mathcal{O}'(X \cup \{s\}, Y)$ consider the two operators. For $\mathcal{O} = \mathcal{U}_\Xi$ just consider that for any Z with $X \cup \{s\} \subseteq Z \subseteq Y$ it holds that $Z \models \varphi_s = s$. The case for $\mathcal{O} = \mathcal{G}_\Xi$ is similar. The set $\{s\}$ is a subset of X, no parent of s is assigned undecided in the pair $(X \cup \{s\}, Y)$ and the set $\{s\}$ satisfies φ_s.*

The proof that $(X, Y \setminus \{s\})$ is admissible for $\mathcal{O}$ proceeds analogous, just consider that for any Z with $X \subseteq Z \subseteq Y \setminus \{s\}$ it holds that $Z \not\models \varphi_s$.

This means that for any admissible pair which assigns undecided to s, there exists a strictly more informative admissible pair such that s is assigned true or false. Therefore s cannot be assigned undecided in a preferred pair, since this pair would not be $\leq_i$-maximal. □

We now define reductions used in multiple proofs as well as showing some properties of interest. The reductions are defined as functions taking sets of (propositional) variables and a formula and mapping them to an ADF. We generally use the sets P, Q, R for propositional variables and use x, y, z as "gadget" statements in the constructed ADF. Without loss of generality we assume that $\{x, y, z\} \cap (P \cup Q \cup R) = \emptyset$. As usual, links of the ADFs are defined implicitly.

Reduction 3.1. Let ψ be a propositional formula over the vocabulary P. Define $\mathsf{RED_1}(P, \psi) = (P \cup \{z\}, L, C)$ with

- $\varphi_p = \neg p$ for $p \in P$; and
- $\varphi_z = \psi$.

This simple ADF can be used to decide satisfiability of ψ or determining if ψ is a tautology using one of the relevant operators.

Lemma 3.13. *Let ψ be a propositional formula over the vocabulary P and $\Xi = \mathsf{RED_1}(P, \psi)$. Further let $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. We find that*

1. *for every conflict-free pair (X, Y) of $\mathcal{O}$ it holds that $P \subseteq (Y \setminus X)$,*
2. *$z \in \mathcal{U}'_\Xi(\emptyset, P \cup \{z\})$ iff ψ is a tautology,*
3. *$z \in \mathcal{O}''(\emptyset, P \cup \{z\})$ iff ψ is satisfiable,*
4. *$z \notin \mathcal{O}''(\emptyset, P)$ iff ψ is unsatisfiable,*
5. *$(\{z\}, P \cup \{z\})$ is conflict-free for $\mathcal{O}$ iff ψ is satisfiable,*
6. *$\mathcal{O}(\emptyset, P \cup \{z\}) = \mathcal{O}(\emptyset, P)$.*

Proof. *Note that $\mathcal{U}''_\Xi = \mathcal{G}''_\Xi$ by Lemma 3.1.*

1. *The first item follows immediately from Lemma 3.9.*
2. *The formula ψ is defined over the vocabulary P. By definition we have $z \in \mathcal{U}'_\Xi(\emptyset, P \cup \{z\})$ iff for all Z with $\emptyset \subseteq Z \subseteq P \cup \{z\}$ we have $Z \models \varphi_z = \psi$. Clearly if $z \in \mathcal{U}'_\Xi(\emptyset, P \cup \{z\})$, then all two-valued interpretations over P are then satisfying assignments of ψ, thus ψ is a tautology in this case. For the converse direction assume that ψ is a tautology. Then for all Z with $\emptyset \subseteq Z \subseteq P$ we have $Z \models \psi$. Since z does not occur in $\varphi_z = \psi$ we know that for all Z with $\emptyset \subseteq Z \subseteq P \cup \{z\}$ it holds that $Z \models \psi$ and thus $z \in \mathcal{U}'_\Xi(\emptyset, P \cup \{z\})$.*

3. *If $z \in \mathcal{O}''(\emptyset, P \cup \{z\})$, then by definition we can infer that there exists a Z such that $\emptyset \subseteq Z \subseteq P \cup \{z\}$ and $Z \models \psi$. Then $Z \setminus \{z\}$ is a satisfying assignment of ψ, therefore ψ is satisfiable. For the other direction assume that ψ is satisfiable. Then there exists a Z with $\emptyset \subsetneq Z \subseteq P$ and $Z \models \psi$. Clearly we have $Z \subseteq P \cup \{z\}$ and thus it holds that $z \in \mathcal{O}''(\emptyset, P \cup \{z\})$.*
4. *Follows analogously as the previous item. Note that z is not in the vocabulary of ψ, and if $z \notin \mathcal{O}''(\emptyset, P)$, then by definition we know that for all Z with $\emptyset \subseteq Z \subseteq P$ it holds that $Z \not\models \varphi_z = \psi$. For the other direction, if ψ is unsatisfiable, then for all Z with $\emptyset \subseteq Z \subseteq P$ it holds that $Z \not\models \psi$, and thus $z \notin \mathcal{O}''(\emptyset, P)$.*
5. *The pair $(\{z\}, P\cup\{z\})$ is conflict-free for $\mathcal{O}$ iff $\{z\} \subseteq \mathcal{O}''(\{z\}, P\cup\{z\})$ and $\mathcal{O}'(\{z\}, P\cup\{z\}) \subseteq P \cup \{z\}$. The latter is trivially true. The former holds iff ψ is satisfiable as shown in the third item proven above.*
6. *Lastly, $\mathcal{O}(\emptyset, P \cup \{z\}) = \mathcal{O}(\emptyset, P)$ holds since z does not occur in any acceptance condition.* □

Note that the lemma even implies that $(\{z\}, P\cup\{z\})$ is naive for $\mathcal{O}$ iff ψ is satisfiable, since the elements of P are undecided in any conflict-free pair. The following more involved construction incorporates several techniques we introduced above.

Reduction 3.2. Let ψ be a propositional formula over the vocabulary $P\cup Q$. Define $\mathsf{RED}_2(P, Q, \psi) = (P \cup \text{-}P \cup Q \cup \{z\}, L, C)$ with $\text{-}P = \{\text{-}p \mid p \in P\}$ and

- $\varphi_p = \neg z \wedge \neg\text{-}p$ for $p \in P$;
- $\varphi_{\text{-}p} = \neg z \wedge \neg p$ for $\text{-}p \in \text{-}P$;
- $\varphi_q = \neg q$ for $q \in Q$; and
- $\varphi_z = \neg z \wedge \neg\psi$.

An ADF constructed by RED_2 has a non-trivial admissible pair with respect to both operators iff the quantified Boolean formula (QBF) $\exists P \forall Q \psi$ is true.

Lemma 3.14. *Let ψ be a propositional formula over the vocabulary $P \cup Q$, $\Xi = \mathsf{RED}_2(P, Q, \psi)$ and $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. It holds that*

- *$(\emptyset, P \cup \text{-}P \cup Q \cup \{z\})$ is the only admissible pair of $\mathcal{O}$ iff $\exists P \forall Q \psi$ is false,*
- *there exists an admissible pair (X, Y) of $\mathcal{O}$ with $z \in (S \setminus Y)$ iff $\exists P \forall Q \psi$ is true.*

Proof. Consider an arbitrary admissible pair (X, Y) of $\mathcal{O}$. By Lemma 3.9 we know that for all $q \in Q$ it holds that $q \in (Y \setminus X)$. Due to Lemma 3.11 we know that if $z \in (Y \setminus X)$ then $p, \text{-}p \in (Y \setminus X)$ for all $p \in P$ and $\text{-}p \in \text{-}P$. Due to Lemma 3.10 we know that z is never assigned true in an admissible pair. Therefore if there exists a non-trivial admissible pair for $\mathcal{O}$, then z must be assigned to false in such a pair. Also, due to Lemma 3.1 we have $\mathcal{U}''_\Xi = \mathcal{G}''_\Xi$.

Assume the QBF $\exists P \forall Q \psi$ is true. Then there exists a $P' \subseteq P$ such that for all $Q' \subseteq Q$ we have $P' \cup Q' \models \psi$. Let $M = P' \cup \{\text{-}p \mid p \in P \setminus P'\}$. We now show that the pair $(M, M \cup Q)$ is admissible for $\mathcal{O}$. It is easy to see that for $p \in P$ and $\text{-}p'$ with $p' \in P \setminus P'$ we have $p, \text{-}p' \in \mathcal{O}'(M, M \cup Q)$. The corresponding acceptance conditions are satisfied by M since $z \notin M$ and the complementary statements (p and $\text{-}p$) are assigned a complementary truth value. Also, $(M, M \cup Q)$ assigns all variables in the respective acceptance conditions a truth value different from undecided. Further $(P \setminus P') \cap \mathcal{O}''(M, M \cup Q) = \emptyset$ and $\{\text{-}p \mid p \in P'\} \cap \mathcal{O}''(M, M \cup Q) = \emptyset$, since the corresponding *acceptance conditions evaluate to false under all interpretations between the lower bound M and upper bound $M \cup Q$. It remains to show that $z \notin \mathcal{O}''(M, M \cup Q)$. By assumption we know that for any Z with $M \subseteq Z \subseteq M \cup Q$ we have $Z \models \psi$ (since $P' = M \cap P$ and $M \cap Q = \emptyset$). Therefore $Z \not\models \varphi_z$. Thus $z \notin \mathcal{O}''(M, M \cup Q)$.*

Now we show the other direction. Assume that there exists a non-trivial admissible pair (X, Y) for $\mathcal{O}$. By the observations in the beginning of the proof, we can directly infer two important facts: (i) $z \in (S \setminus Y)$, and (ii) $q \in (Y \setminus X)$ for all $q \in Q$. By admissibility and (i), it follows that $z \notin \mathcal{O}''(X, Y)$ and thus for all Z with $X \subseteq Z \subseteq Y$ we have $Z \not\models \varphi_z$. Since $z \notin Y$ this means that $Z \not\models \neg\psi$ and thus $Z \models \psi$. Let $P' = X \cap P$. We now consider the two-valued valuations assigning P' to true, $P \setminus P'$ to false and any two-valued assignment on Q. Consider Z with $P' \subseteq Z \subseteq (P' \cup Q)$. For all such Z it holds that there is a Z' with $X \subseteq Z' \subseteq Y$ and $Z \cup X = Z'$. Since $Z' \models \psi$, then also $Z \cup X \models \psi$ and $Z \models \psi$ ($X \setminus Z$ contains no variables of ψ, see also (ii)). Therefore for all $Q' \subseteq Q$ we have $P' \cup Q' \models \psi$ and thus $\exists P \forall Q \psi$ is true.

The second item in the Lemma follows directly, since if the QBF is true then z is false in an admissible pair of $\mathcal{O}$ and vice versa. □

While the previously introduced reductions are mostly used to show hardness results, we also use reductions for membership results. One general core construction is below.

Reduction 3.3. Let Ξ be an ADF. Assume that $S = \{s_1, \ldots, s_n\}$ and set $P = \{t_i, u_i, b_{i,j} \mid 1 \leq i, j \leq n\}$. For each statement s_i, the propositional variable t_i indicates that s_i is true, while u_i indicates that s_i is not false. Thus the truth values of the t_i and u_i determine a four-valued interpretation (T, U). The $b_{i,j}$ are used to guess parents that are needed to derive the acceptance of statement s_i in one operator application step; more precisely, $b_{i,j}$ indicates that s_j is a parent of s_i that is "needed" to infer u_i. By φ_i we denote the acceptance formula of s_i; by φ_i^t we denote φ_i where each s_j has been replaced by t_j; by φ_i^b we denote φ_i where each s_j has been replaced by $b_{i,j}$. Now define the formulas (with underlying intuitions on the right)

$$\phi_{T \subseteq U} = \bigwedge_{s_i \in S} (t_i \to u_i) \qquad (T, U) \text{ is a consistent pair}$$

$$\phi_i^{\mathsf{2v}} = \bigwedge_{r_j \in par(s_i)} (u_j \to t_j) \qquad s_i \text{ has no undecided parents}$$

$$\phi_i^{?} = \bigwedge_{r_j \in par(s_i)} ((t_j \to b_{i,j}) \land (b_{i,j} \to u_j)) \qquad \text{guesses for } s_i \text{ are consistent with } (T, U)$$

$$\phi_{\mathsf{fpl}} = \bigwedge_{s_i \in S} (t_i \leftrightarrow (\varphi_i^t \land \phi_i^{\mathsf{2v}})) \qquad \mathcal{G}'_\Xi(T, U) = T$$

$$\phi_{\mathsf{fpu}} = \bigwedge_{s_i \in S} (u_i \leftrightarrow (\varphi_i^b \land \phi_i^{?})) \qquad \mathcal{G}''_\Xi(T, U) = U$$

$$\phi_{\mathsf{cfp}} = \phi_{\mathsf{fpl}} \land \phi_{\mathsf{fpu}} \land \phi_{T \subseteq U} \qquad \mathcal{G}_\Xi(T, U) = (T, U) \text{ and } T \subseteq U$$

Finally, set $\mathsf{RED}_3(\Xi) = \phi_{\mathsf{cfp}}$.

The main property of this encoding is that it correctly captures consistent fixpoints of the approximate operator.

Lemma 3.15. *Let Ξ be an ADF over statements S and $\phi_{\mathsf{cfp}} = \mathsf{RED}_3(\Xi)$.*

1. *From each model of ϕ_{cfp}, we can read off a consistent fixpoint of $\mathcal{G}_\Xi$;*

2. *conversely, for each consistent fixpoint of $\mathcal{G}_\Xi$, there is a model of ϕ_{cfp}.*

Proof. 1. Let $I \subseteq P$ be such that $I \models \phi_{\mathsf{cfp}}$. Define a three-valued pair (T, U) (the associated pair of I) and a sequence $B_1, \ldots, B_n$ by setting

- $s_i \in T$ iff $t_i \in I$ and $s_i \in U$ iff $u_i \in I$, and
- $s_j \in B_i$ iff $b_{i,j} \in I$.

We have to show $T \subseteq U$ and $\mathcal{G}_\Xi(T, U) = (T, U)$.
For the first part, let $s_i \in T$. Then $t_i \in I$ by definition. Since $I \models \phi_{\mathsf{cfp}}$, in particular $I \models \phi_{T \subseteq U}$, that is, $I \models \bigwedge_{s_i \in S}(t_i \to u_i)$. Thus $I \models u_i$ and by definition $s_i \in U$.
For the second part, we have

$$\begin{aligned}
s_i \in \mathcal{G}'_\Xi(T, U) &\text{ iff } T \models \varphi_i \text{ and } par(s_i) \cap U \subseteq par(s_i) \cap T \\
&\text{ iff } I \models \varphi_i^t \text{ and } I \models \phi_i^{\mathsf{2v}} \\
&\text{ iff } I \models \varphi_i^t \wedge \phi_i^{\mathsf{2v}} \\
&\text{ iff } I \models t_i && \text{(since } I \models \phi_{\mathsf{fpl}}) \\
&\text{ iff } s_i \in T
\end{aligned}$$

Hence $\mathcal{G}'_\Xi(T, U) = T$. Similarly, for the upper bound we have

$$\begin{aligned}
s_i \in \mathcal{G}'_\Xi(U, T) &\text{ iff } B_i \models \varphi_i \text{ and } par(s_i) \setminus B_i \subseteq S \setminus T \text{ and } B_i \subseteq U \\
&\text{ iff } I \models \varphi_i^b \text{ and } I \models \bigwedge_{s_j \in S} ((\neg b_{i,j} \to \neg t_j) \wedge (b_{i,j} \to u_j)) \\
&\text{ iff } I \models \varphi_i^b \text{ and } I \models \bigwedge_{s_j \in S} ((t_j \to b_{i,j}) \wedge (b_{i,j} \to u_j)) \\
&\text{ iff } I \models \varphi_i^b \text{ and } I \models \phi_i^{?} \\
&\text{ iff } I \models (\varphi_i^b \wedge \phi_i^{?}) && \text{(since } I \models \phi_{\mathsf{fpu}}) \\
&\text{ iff } I \models u_i \\
&\text{ iff } s_i \in U
\end{aligned}$$

Hence $U = \mathcal{G}'_\Xi(U, T)$ and in combination $\mathcal{G}_\Xi(T, U) = (T, U)$.

2. Let $\mathcal{G}_\Xi(T, U) = (T, U)$ with $T \subseteq U$. Define an interpretation $I \subseteq P$ as follows:

- Set $t_i \in I$ iff $s_i \in T$ and $u_i \in I$ iff $s_i \in U$.
- Since $\mathcal{G}'_\Xi(U, T) = U$, we have for each $1 \leq i \leq n$ that $s_i \in U$ iff there is a $B_i \subseteq par(s_i)$ with $B_i \models \varphi_i$, $par(s_i) \setminus B_i \subseteq S \setminus T$ and $B_i \subseteq U$. Now pick such a B_i for each $s_i \in S$ and set $b_{i,j} \in I$ iff $s_j \in B_i$.

We have to show $I \models \phi_{\mathsf{cfp}}$. Since $T \subseteq U$, it is clear that $I \models \phi_{T \subseteq U}$ since for all i, $s_i \in T$ implies $s_i \in U$. For the operator applications, we get, for any $s_i \in S$,

$$\begin{aligned}
I \models t_i &\text{ iff } s_i \in T \\
&\text{ iff } s_i \in \mathcal{G}'_\Xi(T, U) \\
&\text{ iff } T \models \varphi_i \text{ and } par(s_i) \cap U \subseteq par(s_i) \cap T \\
&\text{ iff } I \models \varphi_i^t \text{ and } I \models \phi_i^{\mathsf{2v}} \\
&\text{ iff } I \models \varphi_i^t \wedge \phi_i^{\mathsf{2v}}
\end{aligned}$$

Thus $I \models \phi_{\mathsf{fpl}}$. For the upper bound, for any $s_i \in S$,

$$
\begin{aligned}
I \models u_i \text{ iff } & s_i \in U \\
\text{iff } & s_i \in \mathcal{G}'_\Xi(U,T) \\
\text{iff } & B_i \models \varphi_i \text{ and } par(s_i) \setminus B_i \subseteq S \setminus T \text{ and } B_i \subseteq U \\
\text{iff } & I \models \varphi_i^b \text{ and } I \models \bigwedge_{s_j \in S} ((\neg b_{i,j} \to \neg t_j) \land (b_{i,j} \to u_j)) \\
\text{iff } & I \models \varphi_i^b \text{ and } I \models \bigwedge_{s_j \in S} ((t_j \to b_{i,j}) \land (b_{i,j} \to u_j)) \\
\text{iff } & I \models \varphi_i^b \text{ and } I \models \phi_i^?
\end{aligned}
$$

Hence $I \models \phi_{\mathsf{fpu}}$ and in total $I \models \phi_{\mathsf{fpl}} \land \phi_{\mathsf{fpu}} \land \phi_{T \subseteq U}$.

□

3.6. Operator complexities

We next analyze the computational complexity of deciding whether a single statement is contained in the lower or upper bound of the revision of a given pair. This then leads to the complexity of checking whether current lower/upper bounds are pre- or postfixpoints of the revision operators for computing new lower/upper bounds, that is, whether the revisions represent improvements in terms of the information ordering. Intuitively, these results describe how hard it is to "use" the operators and lay the foundation for the rest of the complexity results. Formally we express these notions via the decision problems $\mathsf{Elem}^{\mathcal{I}'}$ and $\mathsf{RevBound}_\circ^{\mathcal{I}'}$ with $\circ \in \{\subseteq, \supseteq\}$, respectively with $\mathcal{I}''$ in the superscript. Recall that $\mathsf{Elem}^{\mathcal{I}'}$ ($\mathsf{Elem}^{\mathcal{I}''}$) denotes the decision problem of verifying if a given element (statement) is contained in the revision of the lower (upper) bound of a given operator and pair. With $\mathcal{I}'$ we denote the operators for revising the lower bound and with $\mathcal{I}''$ the operators for revising the upper bound. The problem $\mathsf{RevBound}_\circ^{\mathcal{I}'}$ asks whether for a given pair (X, Y) we can compare a given set B via $\circ$ with the revised lower bound of this pair. For instance $\mathsf{RevBound}_\supseteq^{\mathcal{G}'}$ denotes the problem of verifying that for a given (X, Y), $B \subseteq S$ and $\mathcal{G}_\Xi \in \mathcal{G}$ we have $\mathcal{G}'_\Xi(X, Y) \subseteq B$.

Proposition 3.16. *Let $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. It holds that*
1. $\mathsf{Elem}^{\mathcal{G}'}$ *is in* P,
2. $\mathsf{Elem}^{\mathcal{U}'}$ *is* coNP*-complete,*
3. $\mathsf{Elem}^{\mathcal{I}''}$ *is* NP*-complete.*

Proof. Let Ξ be an ADF, $s \in S$ and $X \subseteq Y \subseteq S$.
1. $\mathsf{Elem}^{\mathcal{G}'}$ *is in* P*: Since $X \subseteq Y$, we have that whenever there exists a $B \subseteq X \cap par(s)$ with $C_s(B) = \mathbf{t}$ and $par(s) \setminus B \subseteq S \setminus Y$, we know that $B = X \cap par(s)$: Assume there is an $r \in (X \cap par(s)) \setminus B$. Then $r \in par(s)$ and $r \notin B$, whence $r \in par(s) \setminus B \subseteq S \setminus Y$. By $r \in X \subseteq Y$ we get $r \notin S \setminus Y$, contradiction. Thus $B = X \cap par(s)$. Now*

$$
\begin{aligned}
s \in \mathcal{G}'_\Xi(X,Y) \text{ iff } & \text{there exists } B \subseteq X \cap par(s) \text{ with } C_s(B) = \mathbf{t} \text{ and } par(s) \setminus B \subseteq S \setminus Y \\
\text{iff } & C_s(X \cap par(s)) = \mathbf{t} \text{ and } par(s) \setminus X \subseteq S \setminus Y \\
\text{iff } & C_s(X \cap par(s)) = \mathbf{t} \text{ and } (Y \setminus X) \cap par(s) = \emptyset
\end{aligned}
$$

For acceptance functions represented by propositional formulas, $C_s(X \cap par(s)) = \mathbf{t}$ can be decided in polynomial time, since we only have to check whether $X \models \varphi_s$. It can be decided in quadratic time whether there is an undecided parent $r \in par(s)$ with $r \in Y \setminus X$.

2. $\mathsf{Elem}^{\mathcal{U}'}$ *is* coNP*-complete:*

   in coNP: *To decide that* $s \notin \mathcal{U}'_{\sqsubseteq}(X,Y)$*, we guess a* Z *with* $X \subseteq Z \subseteq Y$ *and verify that* $Z \not\models \varphi_s$*.*

   coNP-hard: *We provide a reduction from the problem of determining if a given propositional formula* ψ *is a tautology. Let* ψ *be an arbitrary formula over vocabulary* P*. Construct* $\Xi_\psi = \mathsf{RED}_1(P,\psi)$ *as defined in Reduction 3.1. By Lemma 3.13 it follows that* $z \in \mathcal{U}'_{\sqsubseteq_{\Xi_\psi}}(\emptyset, P \cup \{z\})$ *iff* ψ *is a tautology.*

3. $\mathsf{Elem}^{\mathcal{I}''}$ *is* NP*-complete: Due to Lemma 3.1 we know that* $\mathcal{G}''_{\sqsubseteq}(X,Y) = \mathcal{U}''_{\sqsubseteq}(X,Y)$*.*

   in NP: *To decide that* $s \in \mathcal{U}''_{\sqsubseteq}(X,Y)$*, we guess a* Z *with* $X \subseteq Z \subseteq Y$ *and verify that* $Z \models \varphi_s$*.*

   NP-hard: *For hardness, we provide a reduction from SAT. Let* ψ *be a propositional formula over vocabulary* P*. Construct* $\Xi_\psi = \mathsf{RED}_1(P,\psi)$ *as defined in Reduction 3.1. By Lemma 3.13 it follows that* $z \in \mathcal{U}''_{\sqsubseteq_{\Xi_\psi}}(\emptyset, P \cup \{z\})$ *iff* ψ *is satisfiable.* □

These results can also be formulated in terms of partial evaluations of acceptance formulas: We have $s \in \mathcal{G}'_{\sqsubseteq}(X,Y)$ iff the partial evaluation $\varphi_s^{(X,Y)}$ is a formula without variables that evaluates to $\mathbf{t}$. Similarly, we have $s \in \mathcal{G}''_{\sqsubseteq}(X,Y) = \mathcal{U}''_{\sqsubseteq}(X,Y)$ iff the partial evaluation $\varphi_s^{(X,Y)}$ is satisfiable. Under standard complexity assumptions, computing a new lower bound with the ultimate operator is harder than with the approximate operator. This is because, intuitively, $s \in \mathcal{U}'_{\sqsubseteq}(X,Y)$ iff the partial evaluation $\varphi_s^{(X,Y)}$ is a tautology. The results for Elem straightforwardly lead to the complexity of revising lower/upper bounds for both operators. Note that the results depend crucially on restricting revision to *consistent* pairs (X,Y) (those with $X \subseteq Y$) – for otherwise we could apply $\mathcal{G}''_{\sqsubseteq}(X,Y) = \mathcal{G}'_{\sqsubseteq}(Y,X)$ and use the polynomial-time computable approximate lower bound operator $\mathcal{G}'_{\sqsubseteq}$ on an inconsistent pair (Y,X) to compute $\mathcal{G}'_{\sqsubseteq}(Y,X) = \mathcal{G}''_{\sqsubseteq}(X,Y)$.

Lemma 3.17. *Let* $\mathcal{I} \in \{\mathcal{G},\mathcal{U}\}$ *and* $B \in \{L,U\}$*. It holds that*

1. $\mathsf{RevBound}^{\mathcal{G}'}_{\supseteq}$ *and* $\mathsf{RevBound}^{\mathcal{G}'}_{\subseteq}$ *are in* P*,*
2. $\mathsf{RevBound}^{\mathcal{U}'}_{\supseteq}$ *is in* NP*,*
3. $\mathsf{RevBound}^{\mathcal{U}'}_{\subseteq}$ *is in* coNP*,*
4. $\mathsf{RevBound}^{\mathcal{I}''}_{\subseteq}$ *is in* NP*,*
5. $\mathsf{RevBound}^{\mathcal{I}''}_{\supseteq}$ *is in* coNP*.*

Proof. All results build upon Proposition 3.16. Since the revised lower bound w.r.t. $\mathcal{G}_{\sqsubseteq}$ *can be computed in polynomial time for any ADF* Ξ *we can immediately infer the complexity of the corresponding problems* $\mathsf{RevBound}^{\mathcal{G}'}_{\supseteq}$ *and* $\mathsf{RevBound}^{\mathcal{G}'}_{\subseteq}$*.*

Let $\Xi = (S,L,C)$ *be an ADF,* $\mathcal{O} \in \{\mathcal{G}_{\sqsubseteq},\mathcal{U}_{\sqsubseteq}\}$*,* $B \subseteq S$ *and* $X \subseteq Y \subseteq S$*. Deciding whether* $B \subseteq \mathcal{U}'_{\sqsubseteq}(X,Y)$ *can be decided via* $|B|$ *independent checks for each* $b \in B$ *whether* $b \in \mathcal{U}'_{\sqsubseteq}(X,Y)$*. Each of these are checks in* coNP *and combining them yields again a check in* coNP*. Therefore* $\mathsf{RevBound}^{\mathcal{U}'}_{\subseteq}$ *is in* coNP*. Likewise deciding whether* $B \subseteq \mathcal{O}''(X,Y)$ *can be decided via* $|B|$ *independent checks* $b \in B$*, each of them in* NP*, yielding again a combined problem in* NP*. Thus* $\mathsf{RevBound}^{\mathcal{I}''}_{\subseteq}$ *is in* NP*.*

For $\mathcal{U}'_{\sqsubseteq}(X,Y) \subseteq B$ *we can decide for each* $s \in (S \setminus B)$ *whether* $s \notin \mathcal{U}'_{\sqsubseteq}(X,Y)$*. If this is the case for all* s*, then it holds that* $\mathcal{U}'_{\sqsubseteq}(X,Y) \subseteq B$*. Deciding whether* $s \notin \mathcal{U}'_{\sqsubseteq}(X,Y)$ *holds is a complementary problem to one in* coNP*, thus combining several of them yields a problem in* NP*. This directly shows that* $\mathsf{RevBound}^{\mathcal{U}'}_{\supseteq}$ *is in* NP*. The proof that* $\mathsf{RevBound}^{\mathcal{I}''}_{\supseteq}$ *is in* coNP *proceeds analogously.* □

3.7. Generic upper bounds

We now show generic upper bounds for the computational complexity of the considered problems. This kind of analysis is in the spirit of the results by Dimopoulos et al. [23, Section 4]. The first item is furthermore a straightforward generalization of [20, Theorem 6.13].

Theorem 3.18. *Let $\mathcal{I} \subseteq \mathcal{A}$ be a set of approximation operators, each defined on a CPO on S-subset pairs for some finite set S. Further let $\mathsf{Elem}^{\mathcal{I}'}$ be in Π_i^P and $\mathsf{Elem}^{\mathcal{I}''}$ be in Σ_i^P.*

1. *The least fixpoint of an $\mathcal{O} \in \mathcal{I}$ can be computed in polynomial time with a polynomial number of calls to a Σ_i^P-oracle.*
2. *$\mathsf{Ver}_{cfi}^{\mathcal{I}}$ is in Σ_i^P; $\mathsf{Cred}_{cfi}^{\mathcal{I}}$ is in Σ_i^P;*
3. *$\mathsf{Ver}_{nai}^{\mathcal{I}}$ is in D_i^P; $\mathsf{Cred}_{nai}^{\mathcal{I}}$ is in Σ_i^P;*
4. *$\mathsf{Ver}_{adm}^{\mathcal{I}}$ is in Π_i^P; $\mathsf{Cred}_{adm}^{\mathcal{I}}$ is in Σ_{i+1}^P;*
5. *$\mathsf{Ver}_{com}^{\mathcal{I}}$ is in D_i^P; $\mathsf{Cred}_{com}^{\mathcal{I}}$ is in Σ_{i+1}^P;*
6. *$\mathsf{Ver}_{pre}^{\mathcal{I}}$ is in Π_{i+1}^P; $\mathsf{Cred}_{pre}^{\mathcal{I}}$ is in Σ_{i+1}^P; $\mathsf{Skept}_{pre}^{\mathcal{I}}$ is in Π_{i+2}^P.*

Proof. Let $A = 2^S$ and $\mathcal{O}$ be an approximating operator on $(A^c, \leq_i)$, the consistent CPO of S-subset pairs. Further let $(X, Y) \in A^c$ and $s \in S$.

Using the same line of reasoning as in the proof of Lemma 3.17 we can immediately infer that under the assumptions of the current theorem that $\mathsf{RevBound}_{\supseteq}^{\mathcal{I}'}$ is in Σ_i^P, $\mathsf{RevBound}_{\subseteq}^{\mathcal{I}'}$ is in Π_i^P, $\mathsf{RevBound}_{\subseteq}^{\mathcal{I}''}$ is in Σ_i^P, $\mathsf{RevBound}_{\supseteq}^{\mathcal{I}''}$ is in Π_i^P.

1. For any $(V, W) \in A^c$ we can use the oracle to compute an application of $\mathcal{O}'$ by simply asking whether $z \in \mathcal{O}'(V, W)$ for each $z \in S$. This means we can compute with a linear number of oracle calls the sets $\mathcal{O}'(V, W)$ and $\mathcal{O}''(V, W)$, thus the pair $\mathcal{O}(V, W)$. Hence we can compute the sequence $(\emptyset, S) \leq_i \mathcal{O}(\emptyset, S) \leq_i \mathcal{O}(\mathcal{O}(\emptyset, S)) \leq_i \ldots$ which converges to the least fixpoint of $\mathcal{O}$ after a linear number of operator applications (and thus a polynomial number of oracle calls).
2. $\mathsf{Ver}_{cfi}^{\mathcal{I}}$ is in Σ_i^P, since we can verify if a given pair is conflict-free for a given operator if the lower bound is a positive instance of $\mathsf{RevBound}_{\subseteq}^{\mathcal{I}''}$ and the upper bound a positive instance $\mathsf{RevBound}_{\supseteq}^{\mathcal{I}'}$ together with the pair and operator as input. These are two independent checks in Σ_i^P. For $\mathsf{Cred}_{cfi}^{\mathcal{I}}$ it suffices to verify that the pair $(\{s\}, S)$ is conflict-free. (If $(\{s\}, S)$ is not conflict-free, by Lemma 3.8 there is no conflict-free pair (X, Y) with $(\{s\}, S) \leq_i (X, Y)$.)
3. For $\mathsf{Ver}_{nai}^{\mathcal{I}}$ we first have to decide $\mathsf{Ver}_{cfi}^{\mathcal{I}}$, which can be done in Σ_i^P. To verify that (X, Y) is naive, that is, $\leq_i$-maximal, we do the following: Assume that $Y \setminus X = \{s_1, \ldots, s_m\}$ and construct the pairs $\bar{p}_i = (X \cup \{s_i\}, Y)$ and $\bar{q}_i = (X, Y \setminus \{s_i\})$ for $1 \leq i \leq m$. It follows from Lemma 3.8 that the pair (X, Y) is naive iff none of the $2m$ pairs $\bar{p}_1, \ldots, \bar{p}_m, \bar{q}_1, \ldots, \bar{q}_m$ is conflict-free. Since $\mathsf{Ver}_{cfi}^{\mathcal{I}}$ is in Σ_i^P and the pairs can be verified independently of each other, we need to solve at most $2m \leq 2 \cdot |S|$ independent Σ_i^P problems to show that (X, Y) is not naive. Thus showing that (X, Y) is $\leq_i$-maximal is in Π_i^P, and together with showing conflict-freeness of (X, Y) in Σ_i^P the containment in D_i^P follows. $\mathsf{Cred}_{nai}^{\mathcal{I}}$ coincides with $\mathsf{Cred}_{cfi}^{\mathcal{I}}$ by Lemma 3.7.
4. $\mathsf{Ver}_{adm}^{\mathcal{I}}$ is in Π_i^P, since we can verify if a given pair is admissible for a given operator if the lower bound is a positive instance of $\mathsf{RevBound}_{\subseteq}^{\mathcal{I}'}$ and the upper bound a positive instance $\mathsf{RevBound}_{\supseteq}^{\mathcal{I}''}$ together with the pair and operator as input. These are two independent checks in Π_i^P. For $\mathsf{Cred}_{adm}^{\mathcal{I}}$, we guess a pair (X_1, Y_1) with $s \in X_1$ and check if it is admissible.

5. $\mathsf{Ver}^{\mathcal{I}}_{com}$ *is in* D^P_i*, since we can verify if a given pair is admissible for a given operator in* Π^P_i*. By determining if the lower bound is a positive instance of* $\mathsf{RevBound}^{\mathcal{I}'}_{\supseteq}$ *and the upper bound a positive instance of* $\mathsf{RevBound}^{\mathcal{I}''}_{\subseteq}$ *together with the pair and operator as input we can infer that the given pair is a fix point of the operator. These are two independent checks in* Σ^P_i*, thus combined yields a check in* D^P_i*.* $\mathsf{Cred}^{\mathcal{I}}_{com} = \mathsf{Cred}^{\mathcal{I}}_{adm}$ *by Lemma 3.5.*
6. *For* $\mathsf{Ver}^{\mathcal{I}}_{pre}$*, we show that the co-problem is in* Σ^P_{i+1}*. To show that* (X, Y) *is not a preferred pair, we can show that (1)* (X, Y) *is not a complete pair, which can be decided in* D^P_i*; or (2) that there is a complete pair* (X_1, Y_1) *with* $(X, Y) <_i (X_1, Y_1)$*, which can be done by guessing* (X_1, Y_1) *and showing in* D^P_i *that* (X_1, Y_1) *is complete.*
   $\mathsf{Cred}^{\mathcal{I}}_{pre}$*: coincides with credulous reasoning w.r.t. admissibility, see Lemma 3.5.*
   $\mathsf{Skept}^{\mathcal{I}}_{pre}$*: Consider the co-problem, i.e. deciding whether there exists a preferred pair* (X_1, Y_1) *with* $X_1 \cap \{a\} = \emptyset$*. We guess such a pair* (X_1, Y_1) *and check if it is preferred.*

Naturally, the capability of solving the functional problem of *computing* the grounded semantics allows us to solve the associated decision problems.

Corollary 3.19. *Under the assumptions of Theorem 3.18, the problems* $\mathsf{Ver}^{\mathcal{I}}_{grd}$ *and* $\mathsf{Exists}^{\mathcal{I}}_{grd}$ *are in* Δ^P_{i+1}*.*

4. Complexity of General ADFs

Due to the coincidence of $\mathcal{G}''_{\Xi}$ and $\mathcal{U}''_{\Xi}$ (Lemma 3.1), the computational complexities of decision problems that concern only the upper bound operator also coincide. This will save both work and space in the subsequent developments. Additionally, for all containment results (except for the grounded semantics and existence of non-trivial approximate conflict-free pairs), we can use Theorem 3.18 and need only show hardness.

4.1. Conflict-free semantics

For an ADF Ξ and an operator $\mathcal{O} \in \{\mathcal{G}_{\Xi}, \mathcal{U}_{\Xi}\}$, a pair (X, Y) is conflict-free by definition if and only if $X \subseteq \mathcal{O}''(X, Y)$ and $\mathcal{O}'(X, Y) \subseteq Y$. For the ultimate operator $\mathcal{U}_{\Xi}$, this intuitively means the following:

- For every statement $s \in X$ that is set to true, its partially evaluated acceptance formula $\varphi_s^{(X,Y)}$ must be satisfiable.
- For every statement $s \in S \setminus Y$ that is set to false, its partially evaluated acceptance formula $\varphi_s^{(X,Y)}$ must be refutable.

So roughly, conflict-freeness dictates that the pair must not make truth value assignments that are completely absurd in that a statement is set to true in the pair although its acceptance formula is unsatisfiable with respect to the pair (or symmetrically set to false while the formula is a tautology).

For the approximate operator $\mathcal{G}_{\Xi}$, the requirement for setting statements to false is weaker than for the ultimate operator. (The requirement for setting statements to true is the same since $\mathcal{G}''_{\Xi} = \mathcal{U}''_{\Xi}$.) For the approximate operator, a statement $s \in S$ can be set to false in a pair (X, Y) as long as it is not the case that the formula $\varphi_s^{(X,Y)}$ is a Boolean expression consisting of truth values and connectives that evaluates to true. Conversely, the statement can be set to false if either (1) the formula $\varphi_s^{(X,Y)}$ is a Boolean expression consisting of truth values and connectives that evaluates to false, or (2) the formula $\varphi_s^{(X,Y)}$ contains variables.

Example 4.1. Consider the ADF $D = (S, L, C)$ with $S = \{a, b\}$ and L and C given by $\varphi_a = \neg a$ and $\varphi_b = a \vee \neg a$. For any pair (X, Y) with $a \in X$, that is, any pair that sets a to true, we have that $\varphi_a^{(X,Y)} = \neg \mathbf{t} \equiv \mathbf{f}$ is unsatisfiable. Thus such a pair is not ultimate conflict-free. Symmetrically, for any pair (X, Y) with $a \notin Y$, we find that $\varphi_a^{(X,Y)} = \neg \mathbf{f} \equiv \mathbf{t}$ is irrefutable, and the pair is also not ultimate conflict-free. So our only chance is to set a to undecided, that is, $a \in Y$ and $a \notin X$. For statement b, we see that $\varphi_b^{(\{b\},\{a,b\})} = a \vee \neg a$ is satisfiable, whence $(\{b\}, \{a, b\})$ is ultimate conflict-free. For the pair $(\emptyset, \{a\})$ where b is false, we see that $\varphi_b^{(\emptyset,\{a\})} = a \vee \neg a$ is a tautology, whence the pair is not an ultimate conflict-free pair. However, $\varphi_b^{(\emptyset,\{a\})} = a \vee \neg a$ is an expression containing variables, whence $(\emptyset, \{a\})$ is an approximate conflict-free pair.

This intuition based on satisfiability and refutability will help us in obtaining complexity results for semantics based on the property of being conflict-free. To begin with, to verify that a pair is conflict-free, we obviously have to solve a combined satisfiability/refutability problem.

Proposition 4.1. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Ver}_{cfi}^{\mathcal{I}}$ is NP-complete.*

Proof. Membership follows from Theorem 3.18. For hardness, we provide a reduction from SAT by Reduction 3.1. Let ψ be a formula over vocabulary P. Let $\Xi = \mathsf{RED}_1(P, \psi)$ and $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. Due to Lemma 3.13 we have that the pair $(\{z\}, P \cup \{z\})$ is conflict-free for $\mathcal{O}$ iff ψ is satisfiable. □

For deciding whether there exists a non-trivial ultimate conflict-free pair, we can reduce the propositional satisfiability problem back and forth.

Proposition 4.2. *$\mathsf{Exists}_{cfi}^{\mathcal{U}}$ is NP-complete.*

Proof. in NP: *We can guess a non-trivial pair (X, Y) and the witnesses that verify conflict-freeness of (X, Y) in one sequence of independent guesses. More formally, we reduce $\mathsf{Exists}_{cfi}^{\mathcal{U}}$ to SAT. Let $\Xi = (S, L, C)$ be an ADF. We define a formula φ_Ξ such that $\mathcal{U}_\Xi$ has a non-trivial conflict-free pair iff φ_Ξ is satisfiable.*

Assume $S = \{s_1, \ldots, s_n\}$ and define the vocabulary of φ_Ξ as

$$P = \left\{ s_i^{\mathbf{t}}, s_i^{\mathbf{f}}, p_{i,j} \;\middle|\; 1 \leq i, j \leq n \right\}$$

For $1 \leq i, j \leq n$, denote by φ_i the acceptance formula φ_{s_i} where each occurrence of s_j has been replaced by $p_{j,i}$. Intuitively, atom $p_{j,i}$ is used to guess a truth value for s_j in the acceptance formula of s_i. Now define

$$\varphi_{cfi}^{\mathbf{t},i} = s_i^{\mathbf{t}} \to \left(\varphi_i \wedge \bigwedge_{1 \leq j \leq n} p_{i,j} \right)$$

$$\varphi_{cfi}^{\mathbf{f},i} = s_i^{\mathbf{f}} \to \left(\neg\varphi_i \wedge \bigwedge_{1 \leq j \leq n} \neg p_{i,j} \right)$$

$$\varphi_{cfi} = \bigwedge_{1 \leq i \leq n} \left(\varphi_{cfi}^{\mathbf{t},i} \wedge \varphi_{cfi}^{\mathbf{f},i} \right)$$

$$\varphi_{\mathsf{nt}} = \bigvee_{1 \leq i \leq n} \left(s_i^{\mathbf{t}} \vee s_i^{\mathbf{f}} \right)$$

$$\varphi_\Xi = \varphi_{cfi} \wedge \varphi_{\mathsf{nt}}$$

We have to show that $\mathcal{U}_\Xi$ has a non-trivial conflict-free pair iff φ_Ξ is satisfiable.

if: *Let $I \subseteq P$ be a model for φ_Ξ. Define the pair (X, Y) by*

$$X = \left\{ s_i \mid s_i^{\mathbf{t}} \in I, 1 \leq i \leq n \right\}$$
$$Y = \left\{ s_i \mid s_i^{\mathbf{f}} \notin I, 1 \leq i \leq n \right\}$$

Since in particular I is a model for φ_{nt}, there is an $i \in \{1, \dots, n\}$ such that $s_i \in X$ or $s_i \notin Y$, that is, (X, Y) is non-trivial. It remains to show that (X, Y) is conflict-free.

$X \subseteq \mathcal{U}''_\Xi(X, Y)$: *Let $s_i \in X$. By definition $s_i^{\mathbf{t}} \in I$ and thus I is a model for φ_i. Define $J = \{s_j \mid p_{j,i} \in I\}$. For each $1 \leq k \leq n$, we have that $s_k \in X$ implies $s_k^{\mathbf{t}} \in I$ and thus $p_{k,i} \in I$, whence $s_k \in J$; likewise $s_k \notin Y$ implies $s_k^{\mathbf{f}} \in I$ and thus $p_{k,i} \notin I$ whence $s_k \notin J$. Now I is a model for φ_i, thus J is a model for $\varphi_{s_i}^{(X,Y)}$.*

$\mathcal{U}'_\Xi(X, Y) \subseteq Y$: *Let $s_i \in S \setminus Y$. By definition $s_i^{\mathbf{f}} \in I$ and I falsifies φ_i. As above, we can define J and show that it is compatible with (X, Y). Consequently, J falsifies $\varphi_{s_i}^{(X,Y)}$.*

only if: *Let (X, Y) be a non-trivial conflict-free pair. Define an interpretation I of P as follows. For each $s_i \in X$ set $s_i^{\mathbf{t}}$ to true, $s_i^{\mathbf{f}}$ to false, and $p_{i,j}$ to true for all $1 \leq j \leq n$; for each $s_i \in S \setminus Y$ set $s_i^{\mathbf{f}}$ to true, $s_i^{\mathbf{t}}$ to false, and $p_{i,j}$ to false for all $1 \leq j \leq n$. (There exists at least one such s_i since (X, Y) is non-trivial.) Now let $s_k \in S$.*

- *If $s_k \in X$, we have $s_k \in \mathcal{U}''_\Xi(X, Y)$, whence $\varphi_{s_k}^{(X,Y)}$ is satisfiable. Let I_k be a model for $\varphi_{s_k}^{(X,Y)}$ and note that $I_k \subseteq Y \setminus X$. Now for each $s_j \in Y \setminus X$, set $p_{j,k} \in I$ iff $s_j \in I_k$. Clearly I satisfies φ_k by construction. Since we already defined $p_{k,j}$ to be true for all $1 \leq j \leq n$, we find that I is a model for $\varphi_{\mathit{cfi}}^{\mathbf{t},k}$. Since $s_k^{\mathbf{f}}$ is false, I is also a model for $\varphi_{\mathit{cfi}}^{\mathbf{f},k}$.*
- *If $s_k \in S \setminus Y$, we have $s_k \notin \mathcal{U}'_\Xi(X, Y)$, whence $\varphi_{s_k}^{(X,Y)}$ is refutable. As above, let I_k be a falsification of $\varphi_{s_k}^{(X,Y)}$ and for $s_j \in Y \setminus X$ set $p_{j,k} \in I$ iff $s_j \in I_k$. Then I falsifies φ_k. Again, I is a model for $\varphi_{\mathit{cfi}}^{\mathbf{f},k}$ and $\varphi_{\mathit{cfi}}^{\mathbf{t},k}$.*
- *If $s_k \in Y \setminus X$, we set $s_k^{\mathbf{t}}$ and $s_k^{\mathbf{f}}$ to false in I and for $s_j \in Y \setminus X$ the $p_{j,k}$ arbitrary in I. Clearly I is a model for $\varphi_{\mathit{cfi}}^{\mathbf{t},k}$ and $\varphi_{\mathit{cfi}}^{\mathbf{f},k}$.*

Hence I is a model for φ_{cfi}; since (X, Y) is non-trivial, I is also a model of φ_{nt}. Thus I is a model for φ_Ξ.

NP-hard: *We provide a reduction from SAT. Let ψ be a propositional formula over vocabulary $P \neq \emptyset$. Define an ADF $\Xi_\psi = (S, L, C)$ with $S = P \cup \{z\}$ (where $z \notin P$), $\varphi_p = \neg p$ for $p \in P$ and $\varphi_z = \neg z \wedge \neg\psi$. We have to show that $\mathcal{U}_{\Xi_\psi}$ has a non-trivial conflict-free pair iff ψ is satisfiable.*

if: *If ψ is satisfiable, then $\neg\psi$ is refutable and $\varphi_z^{(\emptyset,P)} = \neg\mathbf{f} \wedge \neg\psi$ is refutable. Thus $z \notin \mathcal{U}'_{\Xi_\psi}(\emptyset, P)$, whence $\emptyset \subseteq \mathcal{U}''_{\Xi_\psi}(\emptyset, P)$ and $\mathcal{U}'_{\Xi_\psi}(\emptyset, P) \subseteq P$ and $(\emptyset, P)$ is non-trivial ($z \notin P$) and conflict-free.*

only if: *Let (X, Y) be conflict-free and non-trivial. Then $X \neq \emptyset$ or $Y \subsetneq S$. Clearly conflict-freeness implies that $P \cap X = \emptyset$ and $P \subseteq Y$. So $z \in X$ or $z \notin Y$. If $z \in X$ then $(X, Y) = (\{z\}, S)$ and by conflict-freeness $z \in \mathcal{U}''_{\Xi_\psi}(\{z\}, S)$. Then $\varphi_z^{(\{z\},S)} = \neg\mathbf{t} \wedge \neg\psi$ is satisfiable, contradiction. Thus $X = \emptyset$ and $z \notin Y$, that is, $(X, Y) = (\emptyset, P)$. Since $(\emptyset, P)$ is conflict-free, $z \notin \mathcal{U}'_{\Xi_\psi}(\emptyset, P)$. Thus $\varphi_z^{(\emptyset,P)} = \neg\mathbf{f} \wedge \neg\psi$ is refutable, that is, $\neg\psi$ is refutable whence ψ is satisfiable.* □

Since there exists a non-trivial conflict-free pair if and only if there exists a non-trivial naive pair (Lemma 3.7), we have this easy consequence.

Corollary 4.3. $\mathsf{Exists}^{\mathcal{U}}_{nai}$ *is* NP*-complete.*

Fortunately, credulous reasoning over conflict-free pairs is not harder than just guessing a pair where the desired statement is true.

Proposition 4.4. *Consider any* $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$*.* $\mathsf{Cred}^{\mathcal{I}}_{cfi}$ *is* NP*-complete.*

Proof. in NP: *Let* $\Xi = (S, L, C)$ *be an ADF with* $S = \{s_1, \ldots, s_n\}$ *and* $1 \leq k \leq n$*. Intuitively, we can guess a pair* (X, Y) *with* $s_k \in X$ *along with the witnesses showing that* (X, Y) *is conflict-free. More formally, we reduce the problem to SAT. For the ultimate operator, we can adapt the construction of Proposition 4.2. We use the formula* φ_{cfi} *as above and define* $\varphi_{\Xi} = \varphi_{cfi} \wedge s^{\mathbf{t}}_k$*. As above, we can show that there is a conflict-free pair* (X, Y) *with* $s_k \in X$ *if and only if* φ_{Ξ} *is satisfiable.*

NP-hard: *We provide a reduction from SAT. Let* ψ *be a propositional formula over vocabulary* P*. Define an ADF* $\Xi = \mathsf{RED}_1(P, \psi)$ *as in Reduction 3.1. Due to Lemma 3.13 we know that there is a conflict-free pair* (X, Y) *with* $z \in X$ *if and only if* ψ *is satisfiable.* □

Again, Lemma 3.7 yields the same complexity for the naive semantics.

Corollary 4.5. *Consider any* $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$*.* $\mathsf{Cred}^{\mathcal{I}}_{nai}$ *is* NP*-complete.*

To verify that a given pair is naive, we have some more work to do. Recalling that a pair is naive iff it is conflict-free and $\leq_i$-maximal with respect to being conflict-free, we can see that to verify naivety we have to verify conflict-freeness (in NP) and verify that there is no properly $\leq_i$-greater conflict-free pair (in coNP).

Proposition 4.6. *Let* $\mathcal{O} \in \{\mathcal{G}, \mathcal{U}\}$*.* $\mathsf{Ver}^{\mathcal{O}}_{nai}$ *is* DP*-complete.*

Proof. Containment follows from Theorem 3.18, so it suffices to show DP*-hardness. We use a reduction from SAT-UNSAT. Let* (ϕ, ψ) *be a tuple of propositional formulas over vocabularies* P_1 *and* P_2*, respectively, with* $P_1 \cap P_2 = \emptyset$*. Construct an ADF* D *as follows:*

$$
\begin{aligned}
S &= P_1 \cup P_2 \cup \{x, y, z\} \\
\varphi_p &= \neg p \text{ for } p \in P_1 \cup P_2 \\
\varphi_x &= \phi \wedge \neg z \\
\varphi_y &= \psi \vee z \\
\varphi_z &= z
\end{aligned}
$$

Furthermore, define the pair $\bar{n} = (\{x, y\}, S)$*. We now show that* $\bar{n}$ *is naive for* $\mathcal{O}$ *iff* ϕ *is satisfiable and* ψ *is unsatisfiable. (Notice that the proof only uses* $\mathcal{O}''$ *and thus works for* $\mathcal{O} = \mathcal{G}$ *and* $\mathcal{O} = \mathcal{U}$*.)*

if: *Let* ϕ *be satisfiable and* ψ *be unsatisfiable. We show that* $\bar{n}$ *is naive for* $\mathcal{O}$*. We first show that* $\bar{n}$ *is conflict-free for* $\mathcal{O}$*:*

- $\varphi^{\bar{n}}_x = \phi \wedge \neg z$ *is satisfiable by presumption whence* $s \in \mathcal{O}''(\bar{n})$*;*
- $\varphi^{\bar{n}}_y = \psi \vee z$ *is satisfiable whence* $s \in \mathcal{O}''(\bar{n})$*.*

It remains to show that all $\bar{o}$ with $\bar{n} <_i \bar{o}$ are not conflict-free. Clearly, setting a $p \in P_1 \cup P_2$ to true or false in $\bar{n}$ violates conflict-freeness. Furthermore, x and y are already set, so the only two choices are setting z to true or false. If we set z to true, that is, consider the pair $(\{x, y, z\}, S)$, then we observe that the formula $\varphi_x^{(\{x,y,z\},S)} = \phi \wedge \neg \mathbf{t}$ is unsatisfiable whence $x \notin \mathcal{O}''(\{x, y, z\}, S)$ and the pair is not conflict-free. So the only remaining candidate is setting z to false, that is, the pair $\bar{o} = (\{x, y\}, S \setminus \{z\})$. Since ψ is unsatisfiable and does not mention x, y, z, also $\varphi_y^{\bar{o}} = \psi \vee \mathbf{f}$ is unsatisfiable. Thus $y \notin \mathcal{O}''(\bar{o})$ and $\bar{o}$ is not conflict-free. It follows that $\bar{n}$ is naive for $\mathcal{O}$.

only if: *Let $\bar{n}$ be naive for $\mathcal{O}$. Assume to the contrary that ϕ is unsatisfiable or ψ is satisfiable.*

1. *ϕ is unsatisfiable. Then $\varphi_x^{\bar{n}} = \phi \wedge \neg z$ is unsatisfiable and $x \notin \mathcal{O}''(\bar{n})$. Thus $\bar{n}$ is not conflict-free, in contradiction to it being naive.*
2. *ψ is satisfiable. Define the pair $\bar{o} = (\{x, y\}, S \setminus \{z\})$. Since ψ does not mention x, y, z, also the formula $\varphi_y^{\bar{o}} = \psi \vee \mathbf{f}$ is satisfiable. Then $\bar{o}$ is conflict-free with $\bar{n} <_i \bar{o}$, contradiction.*

Thus ϕ is satisfiable and ψ is unsatisfiable. □

For skeptical reasoning, we can do no better than verifying the absence of a naive pair where the statement in question is not true. The hardness proof proceeds by "laying a trap" for the conflict-free semantics: setting a statement s to true (for example) in a pair (X, Y) can be justified *locally* by the partially evaluated acceptance formula $\varphi_s^{(X,Y)}$ being satisfiable. However, the satisfying assignment for statements from $Y \setminus X$ need not pay respect to what might be dictated by other parts of the framework. The proof makes use of this and employs two new statements y and z with acceptance formulas $\varphi_y = z \to \psi$ for a given QBF matrix ψ and $\varphi_z = \mathbf{t}$. Now setting y to true can be locally justified since φ_y is satisfiable by setting z to false. However, in the case where ψ is unsatisfiable, we cannot set both y *and* z to true, since the respective partial evaluation of φ_y would be unsatisfiable, thereby violating conflict-freeness. Consequentially, this case leads to different kinds of naive pairs (those with $y \mapsto \mathbf{t}$ and $z \mapsto \mathbf{u}$, and those with $y \mapsto \mathbf{u}$ and $z \mapsto \mathbf{t}$). This reasoning is used in the proof below.

Proposition 4.7. *Let $\mathcal{O} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Skept}_{nai}^{\mathcal{O}}$ is Π_2^P-complete.*

Proof. in Π_2^P: *Let $\Xi = (S, L, C)$ be an ADF and $s \in S$. We can guess a pair (X, Y) with $s \notin X$ and verify in* DP *that it is naive.*

Π_2^P-hard: *We provide a reduction from the Π_2^P-complete problem of deciding whether a $QBF_{2,\forall}$-formula is valid. Let $\forall P \exists Q \psi$ be an instance of $QBF_{2,\forall}$-TRUTH where $P, Q \neq \emptyset$ and (w.l.o.g.) ψ mentions at least one $p \in P$ and at least one $q \in Q$. We construct an ADF D_ψ with a special statement z that is true in each naive pair of D_ψ if and only if $\forall P \exists Q \psi$ is true. Define $D_\psi = (S, L, C)$ with*

$$\begin{aligned}
S &= P \cup Q \cup \{y, z\} \text{ with } y, z \notin P \cup Q \\
\varphi_p &= p \text{ for } p \in P \\
\varphi_q &= \neg q \text{ for } q \in Q \\
\varphi_y &= z \to \psi \\
\varphi_z &= \mathbf{t}
\end{aligned}$$

if: *Let $\forall P \exists Q \psi$ be true; then for each $M \subseteq P$, the formula $\psi^{(M, M\cup Q)}$ is satisfiable. In particular, for all $M \subseteq N \subseteq P$, the formula $\psi^{(M,N\cup Q)}$ is satisfiable. We show that for each pair (X,Y) that is conflict-free for D_ψ, we can set z to true without violating conflict-freeness. Since z cannot be set to false by definition, it follows that z is true in every naive pair of D_ψ.*

Let (X,Y) be conflict-free for D_ψ. By definition of φ_z we have $z \in \mathcal{O}'(X,Y) \subseteq Y$. Furthermore, all $q \in Q$ are undefined in (X,Y), that is, $Q \subseteq Y \setminus X$. We have to show that $(X \cup \{z\}, Y)$ is conflict-free for D_ψ. Since $z \in Y$, the pair is consistent.

- *Let $t \in X$. By the assumption that (X,Y) is conflict-free, $\varphi_t^{(X,Y)}$ is satisfiable. If $t \in P$, then $\varphi_t^{(X\cup\{z\},Y)} = \varphi_t^{(X,Y)}$ is satisfiable. The case $t \in Q$ is impossible. If $t = y$, then $\varphi_t^{(X\cup\{z\},Y)} = \varphi_y^{(X\cup\{z\},Y)} = \mathbf{t} \to \psi^{(X,Y)} \equiv \psi^{(X,Y)}$ is satisfiable by assumption. The case $t = z$ is trivial.*
- *Let $f \in S \setminus Y$. If $f \in P$, then $\varphi_f^{(X\cup\{z\},Y)} = \varphi_f^{(X,Y)}$* and *$f \notin \mathcal{O}'(X \cup \{z\}, Y)$ since $f \notin \mathcal{O}'(X,Y)$. The cases $f \in Q$ and $f = z$ are impossible. If $f = y$, then $y \notin \mathcal{O}'(X,Y)$.*

  1. *For $\mathcal{O} = \mathcal{U}_{D_\psi}$, this means that $\varphi_y^{(X,Y)} = (z \to \psi)^{(X,Y)}$ is refutable. Then the formula $\psi^{(X,Y)} = \psi^{(X\cup\{z\},Y)} \equiv \mathbf{t} \to \psi^{(X\cup\{z\},Y)} = \varphi_y^{(X\cup\{z\},Y)}$ is refutable and thus $y \notin \mathcal{U}'_{D_\psi}(X \cup \{z\}, Y)$.*
  2. *For $\mathcal{O} = \mathcal{G}_{D_\psi}$, the formula $\varphi_y^{(X\cup\{z\},Y)} = (z \to \psi)^{(X\cup\{z\},Y)}$ contains variables (ψ contains at least one $q \in Q \subseteq Y \setminus X$ by presumption) whence $y \notin \mathcal{G}'_{D_\psi}(X \cup \{z\}, Y)$.*

only if: *Let $M \subseteq P$ be such that the formula $\psi^{(M,M\cup Q)}$ is unsatisfiable. We show that the pair $(M \cup \{y\}, M \cup Q \cup \{y,z\})$ with $z \notin M$ is naive. Clearly, the pair is conflict-free for statements among $P \cup Q \cup \{z\}$. The formula $\varphi_y^{(M\cup\{y\},M\cup Q\cup\{y,z\})} = z \to \psi^{(M,M\cup Q)}$ is satisfiable (set z to false) whence $y \in \mathcal{O}''(M \cup \{y\}, M \cup Q \cup \{y,z\})$. The $p \in P$ are all set to true or false, and none of the $q \in Q$ can be set to true or false; our only possibility is to set z to true or false. Setting z to false is obviously not conflict-free by definition. For the pair $(M \cup \{y,z\}, M \cup Q \cup \{y,z\})$, we have that*

$$\varphi_y^{(M\cup\{y,z\},M\cup Q\cup\{y,z\})} = \mathbf{t} \to \psi^{(M\cup\{y,z\},M\cup Q\cup\{y,z\})} \equiv \psi^{(M,M\cup Q)}$$

is unsatisfiable whence $z \notin \mathcal{O}''(M \cup \{y,z\}, M \cup Q \cup \{y,z\})$ and the pair is not conflict-free. Therefore, $(M \cup \{y\}, M \cup Q \cup \{y,z\})$ is a naive pair where z is not true. □

Recalling that deciding existence of non-trivial ultimate conflict-free pairs is NP-hard, we can show that deciding existence of non-trivial *approximate* conflict-free pairs is (potentially) easier. The reason lies in the smaller precision of the approximate operator, as witnessed in Example 4.1.

Proposition 4.8. $\mathsf{Exists}^{\mathcal{G}}_{cfi}$ *is in* P.

Proof. Let $\Xi = (S, L, C)$ be an ADF. If $S = \emptyset$ then there is no non-trivial pair at all, so assume $S \neq \emptyset$.

1. *There is an $s \in S$ such that the formula $\varphi_s^{(\emptyset, S\setminus\{s\})}$ contains variables. Then by definition of $\mathcal{G}_\Xi$, we have $s \notin \mathcal{G}'_\Xi(\emptyset, S \setminus \{s\})$ and thus $(\emptyset, S \setminus \{s\})$ is conflict-free and non-trivial.*
2. *For all $s \in S$, the formula $\varphi_s^{(\emptyset, S\setminus\{s\})}$ is an expression consisting only of connectives and truth values. Then for each $s \in S$ the expression $\varphi_s^{(\emptyset, S\setminus\{s\})}$ has a fixed truth value, which can be computed in polynomial time.*

(a) *There is an* $s \in S$ *such that* $\varphi_s^{(\emptyset, S\setminus\{s\})} \equiv \mathbf{f}$*. Then by definition of* $\mathcal{G}_\Xi$*, we have* $s \notin \mathcal{G}'_\Xi(\emptyset, S \setminus \{s\})$ *and thus* $(\emptyset, S \setminus \{s\})$ *is conflict-free and non-trivial.*

(b) *For all* $s \in S$*, we find* $\varphi_s^{(\emptyset, S\setminus\{s\})} \equiv \mathbf{t}$*. Then we consider the truth value of* $\varphi_s^{(\{s\},S)}$ *for all* $s \in S$*, which is computable in polynomial time.*

   i. *There is an* $s \in S$ *for which we find* $\varphi_s^{(\{s\},S)} \equiv \mathbf{t}$*. In particular,* $\varphi_s^{(\{s\},S)}$ *is satisfiable, whence* $s \in \mathcal{G}''_\Xi(\{s\}, S)$*. Thus the pair* $(\{s\}, S)$ *is conflict-free and non-trivial.*

   ii. *For all* $s \in S$*, we find* $\varphi_s^{(\{s\},S)} \equiv \mathbf{f}$*. Then for each* $s \in S$*, we have* $\varphi_s \equiv \neg s$ *and* $(\emptyset, S)$ *is the only conflict-free pair.* □

As usual, Lemma 3.7 yields the same bounds for the existence of non-trivial approximate naive pairs.

Corollary 4.9. $\mathsf{Exists}_{nai}^{\mathcal{G}}$ *is in* P*.*

However, these two results are the only ones where the complexities of the approximate and ultimate operators differ for semantics based on conflict-freeness.

4.2. Admissibility-based semantics

We begin our complexity analysis of admissibility-based semantics by introducing and recalling some basic concepts of these semantics and the corresponding operators. By definition, a pair (X, Y) with $X \subseteq Y$ is admissible for operator $\mathcal{O}$ iff $(X, Y) \leq_i \mathcal{O}(X, Y)$. Since the operators under consideration are $\leq_i$-monotone, we can directly infer that if (X, Y) is admissible then it holds that $\mathcal{O}(X, Y) \leq_i \mathcal{O}(\mathcal{O}(X, Y))$. This means that (i) applying operator $\mathcal{O}$ to an admissible pair yields again an admissible pair, and (ii) iterative applications of $\mathcal{O}$ to an admissible pair (X, Y) always yield pairs (X', Y') such that $(X, Y) \leq_i (X', Y')$. In more detail, (ii) implies that a statement assigned to true or false in (X, Y) will keep this assignment if the operator is applied, while undecided statements may change their assignment. If (X, Y) is a fixpoint of $\mathcal{O}$ (i.e. a complete pair of $\mathcal{O}$), then the application of $\mathcal{O}$ also does not change the undecided statements.

Most complexity results in the following mainly rely on the operator for the upper bound $\mathcal{U}'' = \mathcal{G}''$, which is the same for both approximate and ultimate operators (Lemma 3.1). We also make use of the reductions from Section 3.5.

The first problem we analyze is the verification problem for admissible semantics. As before for most problems we need only show hardness since Theorem 3.18 shows membership.

Proposition 4.10. *Consider any* $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$*.* $\mathsf{Ver}_{adm}^{\mathcal{I}}$ *is* coNP*-complete.*

Proof. We provide a reduction from the problem of deciding whether a given formula ψ *over vocabulary* P *is unsatisfiable. Let ADF* $\Xi = \mathsf{RED}_1(P, \psi)$ *as defined in Reduction 3.1 and* $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$*. The pair* $(\emptyset, P)$ *is admissible for* $\mathcal{O}$ *iff* $z \notin \mathcal{U}''_\Xi(\emptyset, P)$ *iff* ψ *is unsatisfiable due to Lemma 3.13.* □

For verifying if a given pair is complete the complexity increases to DP compared to just checking admissibility. Briefly put, the coNP part decides whether the given pair is a postfixpoint and the additional NP check is used to decide whether the pair is prefixpoint. Together they decide whether the pair is a fixpoint.

Proposition 4.11. *Consider any* $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$*.* $\mathsf{Ver}_{com}^{\mathcal{I}}$ *is* DP*-complete.*

Proof. For $\mathcal{I} = \mathcal{U}$ this was proven in [11, Corollary 7]. However, the following reduction works for both operators.

We provide a reduction from the DP-complete problem of determining whether a given formula ϕ is satisfiable and a given formula ψ is unsatisfiable. Let ϕ and ψ be arbitrary formulas over the disjoint vocabularies P_1 and P_2 respectively. Let $P = P_1 \cup P_2$. Construct the following ADF $\Xi = (P \cup \{y, z\}, L, C)$.

- $\varphi_p = p$ for $p \in P$,
- $\varphi_y = \neg y \wedge \phi$,
- $\varphi_z = \psi$.

Let $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. We now prove that $(\emptyset, P \cup \{y\}) = \mathcal{O}(\emptyset, P \cup \{y\})$ iff ϕ is satisfiable and ψ is unsatisfiable.

Independent of ϕ and ψ we know due to Lemma 3.10 that for any conflict-free pair of $\mathcal{O}$ that y is not assigned to true. Further for all $p \in P$ we have $p \in \mathcal{U}''_\Xi(\emptyset, P \cup \{y\})$, since $\varphi_p^{(\emptyset, P\cup\{y\})} = p$ is satisfiable. Further $p \notin \mathcal{U}'_\Xi(\emptyset, P \cup \{y\})$, since $\varphi_p^{(\emptyset, P\cup\{y\})} = p$ is refutable. Thus $p \notin \mathcal{G}'_\Xi(\emptyset, P \cup \{y\})$, since the ultimate operator is at least as precise than the approximate operator.

Consider the following two cases:

1. Let ϕ be satisfiable. Clearly $\varphi_y^{(\emptyset, P\cup\{y\})}$ is satisfiable and thus $y \in \mathcal{U}''_\Xi(\emptyset, P \cup \{y\})$.
2. Let ψ be unsatisfiable. It follows immediately that $z \notin \mathcal{U}''_\Xi(\emptyset, P \cup \{y\})$.

Therefore $(\emptyset, P \cup \{y\})$ is complete for $\mathcal{O}$ if ϕ is satisfiable and ψ is unsatisfiable.

For the other direction assume that ϕ is unsatisfiable or ψ is satisfiable. For the first case suppose ϕ is unsatisfiable. Then $y \notin \mathcal{U}''_\Xi(\emptyset, P \cup \{y\})$ and $(\emptyset, P \cup \{y\})$ is not complete. For the second case suppose ψ is satisfiable. Then $z \in \mathcal{U}''_\Xi(\emptyset, P \cup \{y\})$ and likewise $(\emptyset, P \cup \{y\})$ is not complete. □

Next, we analyze the complexity of verifying that a given pair is the approximate (ultimate) Kripke-Kleene semantics or grounded pair of an ADF Ξ, that is, the least fixpoint of $\mathcal{G}_\Xi$ ($\mathcal{U}_\Xi$). It turns out that verifying if a given pair is grounded has the same complexity as verifying if the pair is complete. For both operators showing membership is the tricky part. For $\mathcal{G}$ we reduce the steps of the operator computation into propositional logic. In particular we construct two formulas, one we check for satisfiability and the other for unsatisfiability. This gives us an interesting, yet technical proof of membership. For $\mathcal{U}$ this result was shown already in [11, Theorem 6], but the proof was omitted due to space limitations. For sake of completeness we will present here an alternative proof which will be re-used later for query-based reasoning. The membership proof is based on a somewhat more involved guess and check algorithm. This algorithm non-deterministically tries to construct a pair (X, Y), which is equal or more informative than the ultimate grounded pair of a given ADF. If the algorithm successfully constructs such a pair with e.g. $s \notin X$, then we can conclude that s is also not true in the ultimate grounded pair. Since the algorithm is based on guess and check this provides us with a procedure showing that deciding whether a statement is not true in the ultimate grounded pair is a problem in NP. Formally we first prove a technical lemma underlying this algorithm. The following definition specifies a set of pairs called $grd_{\leq_i}(\Xi)$ for an ADF Ξ.

Definition 4.1. Let Ξ be an ADF and $X \subseteq Y \subseteq S$. If it holds that

1. for each $x \in X$ there exists a Z_x s.t. $X \subseteq Z_x \subseteq Y$ and $Z_x \models \varphi_x$,
2. for each $s \in S \setminus Y$ there exists a Z_s s.t. $X \subseteq Z_s \subseteq Y$ and $Z_s \not\models \varphi_s$, and

3. for each $y \in Y \setminus X$ there exist two Z_y, Z'_y s.t. $X \subseteq Z_y \subseteq Y$, $X \subseteq Z'_y \subseteq Y$, $Z_y \models \varphi_x$ and $Z'_y \not\models \varphi_y$,

then let $(X, Y) \in grd_{\leq_i}(\Xi)$.

The set $grd_{\leq_i}(\Xi)$ has the appealing property that the grounded pair of $\mathcal{U}_{\Xi}$ is in this set and all other pairs in the set are more informative than the ultimate grounded pair.

Lemma 4.12. *Let Ξ be an ADF, $(X, Y) \in grd_{\leq_i}(\Xi)$ and $(L, U) = lfp(\mathcal{U}_{\Xi})$. It holds that*

1. *$(L, U) \in grd_{\leq_i}(\Xi)$, and*
2. *$(L, U) \leq_i (X, Y)$.*

Proof. *Let (L, U) be the grounded pair of $\mathcal{U}_{\Xi}$, then it is straightforward to show that $(L, U) \in grd_{\leq_i}(\Xi)$, i.e. that this pair satisfies all three properties of Definition 4.1. For each $l \in L$ it holds that $\varphi_l^{(L,U)}$ is tautological, for $s \in S \setminus U$ it holds that $\varphi_s^{(L,U)}$ is unsatisfiable, and for $u \in U \setminus L$ it holds that $\varphi_u^{(L,U)}$ is satisfiable and refutable. By supposing the contrary one immediately arrives at a contradiction that (L, U) is the grounded pair (recall that the grounded pair is also a fixpoint). If $\varphi_l^{(L,U)}$ is not tautological, then $l \notin \mathcal{U}'_{\Xi}(L, U)$, if $\varphi_s^{(L,U)}$ is not unsatisfiable, then $s \in \mathcal{U}''_{\Xi}(L, U)$ and if $\varphi_u^{(L,U)}$ is a tautology or unsatisfiable, then we have in the former case that $u \in \mathcal{U}'_{\Xi}(L, U)$ and the latter that $u \notin \mathcal{U}''_{\Xi}(L, U)$.*

Assume $(X, Y) \in grd_{\leq_i}(\Xi)$, i.e. the pair satisfies all three properties of Definition 4.1, then we show by induction on $n \geq 1$ that $\mathcal{U}^n_{\Xi}(\emptyset, S) \leq_i (X, Y)$, with the usual meaning of iterative applications of operators, i.e. $\mathcal{U}^n_{\Xi}(X, Y) = \mathcal{U}^{n-1}_{\Xi}(\mathcal{U}_{\Xi}(X, Y))$. Note that there exists an $i \geq 0$ such that $\mathcal{U}^i_{\Xi}(\emptyset, S) = lfp(\mathcal{U}_{\Xi})$. For $n = 1$ and $\mathcal{U}^1_{\Xi}(\emptyset, S) = (L_1, U_1)$ it holds that if $s \in L_1$ then φ_s is a tautology, implying that $s \in X$, since otherwise there would exist a two-valued interpretation which does not satisfy φ_s. This is due to the fact that if $s \notin X$, then by assumption and by Definition 4.1 there would exist a Z with $X \subseteq Z \subseteq Y$, such that $Z \not\models \varphi_s$. The case for $s \in S \setminus U_1$ is symmetric. Now assume the induction hypothesis $(L_n, U_n) = \mathcal{U}^n_{\Xi}(\emptyset, S) \leq_i (X, Y)$ and to show that $\mathcal{U}^{n+1}_{\Xi}(\emptyset, S) \leq_i (X, Y)$ holds consider $\mathcal{U}^{n+1}_{\Xi}(\emptyset, S) = (L_{n+1}, U_{n+1})$. If $s \in L_{n+1} \setminus L_n$ then $\varphi_s^{(L_n,U_n)}$ is tautological, which means that s must be in X. Similarly for the statements set to false. This proves the lemma. □

This leads us to Algorithm 1 for deciding whether a certain statement s is not true in the ultimate grounded pair of an ADF Ξ. This is a non-deterministic algorithm, which guesses a pair (X, Y) with $s \notin X$ along with witnesses with which we can verify that $(X, Y) \in grd_{\leq_i}(\Xi)$. If this is the case then $s \notin L$ with (L, U) the grounded pair of $\mathcal{U}_{\Xi}$ (everything assigned to true in the grounded pair is also true in (X, Y) due to the preceding lemma).

Algorithm 1 Guess & check algorithm for ultimate grounded semantics.

Require: ADF $\Xi = (S, L, C)$, $s \in S$.
Ensure: Return *no* iff s is not true in $lfp(\mathcal{U}_{\Xi})$.
Guess X, Y with $X \subseteq Y \subseteq S$ and $s \notin X$;
Guess for each $t \in Y$ a set $Z_t \subseteq S$; store Z_t in T;
Guess for each $f \in (S \setminus X)$ a set $Z_f \subseteq S$; store Z_f in F;
Verify that for each $Z_t \in T$ we have $Z_t \models \varphi_t$;
Verify that for each $Z_f \in F$ we have $Z_f \not\models \varphi_f$;
If there exists a guess such that all verifications were successful **return** *no*;

Clearly, Algorithm 1 requires at most $2 \cdot |S| + 2$ guesses of sets. If $X = \emptyset$ and $Y = S$ we guess two sets per $x \in S$. Each such guess can be constructed and checked in polynomial time with respect to the size of the input ADF. This algorithm returns *no* if there exists a successful computation path and otherwise terminates without returning *no*. More formally we show that the algorithm is correct in the following lemma.

Lemma 4.13. *Let $\Xi = (S, L, C)$ be an ADF, $s \in S$ and $(L, U) = \mathit{lfp}(\mathcal{U}_\Xi)$. It holds that $s \notin L$ iff non-deterministic Algorithm 1 returns no for input ADF Ξ and statement s.*

Proof. Assume Algorithm 1 returned *no* for Ξ and statement s. Then clearly for the guessed (X, Y) (Line 1) we have $s \notin X$ and $(X, Y) \in \mathit{grd}_{\leq_i}(\Xi)$, since witnesses for all three properties of Definition 4.1 were guessed and successfully verified. By Lemma 4.12 we know that $(L, U) \leq_i (X, Y)$. Thus $s \notin L$.

For the other direction assume that $s \notin L$. Due to Lemma 4.12 we know that $(L, U) \in \mathit{grd}_{\leq_i}(\Xi)$. Therefore the non-deterministic Algorithm 1 can guess (L, U) along with witnesses that $(L, U) \in \mathit{grd}_{\leq_i}(\Xi)$ and successfully verify these witnesses. Thus there exists a guess (a computation path) such that Algorithm 1 returns *no*. □

By Lemma 4.12, we can straightforwardly adapt Algorithm 1 to decide (i.e. return "no")

- whether a statement is not false in the ultimate grounded pair, by replacing the guessed pair in Line 1 with (X, Y) such that $s \in Y$; and
- whether a statement is undecided in the ultimate grounded pair, by replacing the guessed pair in Line 1 with (X, Y) such that $s \in (Y \setminus X)$.

Note that the guess and check algorithm can be adapted to either decide whether a statement is not true or false in the ultimate grounded pair, or to decide whether a statement is undecided in the ultimate grounded pair. This gives complementary membership results w.r.t. complexity regarding the decision problem whether a statement is true or false, or on the other hand, if it is undecided in the ultimate grounded pair.

We now come to the theorem showing the computational complexity of the verification problem of grounded semantics for both operators. The proof uses a reduction introduced earlier – it encodes steps of the approximate operator computation into propositional logic. For the ultimate operator we make use of Algorithm 1 and Lemma 4.13.

Theorem 4.14. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Ver}^{\mathcal{I}}_{\mathit{grd}}$ is DP-complete.*

Proof. *Let Ξ be an ADF and $X \subseteq Y \subseteq S$. We begin the proof for $\mathcal{I} = \mathcal{G}$.*

in DP: *We provide a reduction to SAT-UNSAT by extending the construction of Reduction 3.3. We additionally define the formulas*

$$\phi_{\leq_i} = \bigwedge_{s_i \notin X} \neg t_i \wedge \bigwedge_{s_i \in Y} u_i \qquad (T, U) \leq_i (X, Y)$$

$$\phi_{\geq_i} = \bigwedge_{s_i \in X} t_i \wedge \bigwedge_{s_i \notin Y} \neg u_i \qquad (T, U) \geq_i (X, Y)$$

$$\phi_{=} = \phi_{\leq_i} \wedge \phi_{\geq_i} \qquad (T, U) = (X, Y)$$

$$\phi_{<_i} = \phi_{\leq_i} \wedge \neg\phi_{\geq_i} \qquad (T, U) <_i (X, Y)$$

$$\psi_1 = \phi_{\mathsf{cfp}} \wedge \phi_{=} \qquad \mathcal{G}_\Xi(T, U) = (T, U) \text{ with } T \subseteq U \text{ and } (T, U) = (X, Y)$$

$$\psi_2 = \phi_{\mathsf{cfp}} \wedge \phi_{<_i} \qquad \mathcal{G}_\Xi(T, U) = (T, U) \text{ with } T \subseteq U \text{ and } (T, U) <_i (X, Y)$$

Intuitively, $\phi_{=}$ will be used to force (X, Y) to be a fixpoint of $\mathcal{G}_\Xi$, and $\phi_{<_i}$ will be used to stipulate the existence of a fixpoint with strictly less information than (X, Y).

We claim that (1) ψ_1 is satisfiable iff (X, Y) is a consistent fixpoint of $\mathcal{G}_\Xi$, and (2) ψ_2 is satisfiable iff there is a fixpoint $(T, U) <_i (X, Y)$ of $\mathcal{G}_\Xi$. From this it follows that (ψ_1, ψ_2) is a positive instance of SAT-UNSAT iff (X, Y) is the Kripke-Kleene semantics of Ξ.

1. *ψ_1 is satisfiable iff (X, Y) is a consistent fixpoint of $\mathcal{G}_\Xi$.*
   "if": *Let (X, Y) be a consistent fixpoint of $\mathcal{G}_\Xi$. Set $(T, U) = (X, Y)$, then by Lemma 3.15 there is an interpretation $I \subseteq P$ with $I \models \phi_{\mathsf{cfp}}$. By definition, we also have $I \models \phi_=$, whence $I \models \psi_1$ and ψ_1 is satisfiable.*
   "only if": *Let $\psi_1 = \phi_{\mathsf{cfp}} \wedge \phi_=$ be satisfiable. Then in particular ϕ_{cfp} is satisfiable and by Lemma 3.15 there is an interpretation $I \subseteq P$ such that its associated pair (T, U) is a consistent fixpoint of $\mathcal{G}_\Xi$. Since additionally $I \models \phi_=$, it follows that $(T, U) = (X, Y)$.*
2. *ψ_2 is satisfiable iff there is a fixpoint $(T, U) <_i (X, Y)$ of $\mathcal{G}_\Xi$.*
   "if": *Let $(T, U) <_i (X, Y)$ with $T \subseteq U$ and $\mathcal{G}_\Xi(T, U) = (T, U)$. By Lemma 3.15 we can define a two-valued interpretation $I \subseteq P$ such that $I \models \phi_{\mathsf{cfp}}$. It is straightforward to show that $(T, U) <_i (X, Y)$ implies $I \models \phi_{<_i}$.*
   "only if": *Let $I \subseteq P$ be an interpretation with $I \models \psi_2$. Since in particular $I \models \phi_{\mathsf{cfp}}$, Lemma 3.15 yields a consistent fixpoint (T, U) of $\mathcal{G}_\Xi$. As above, we can show that $(T, U) <_i (X, Y)$.*

DP-hard: *This follows from the proof in Proposition 4.11: The complete pair to verify there coincides with the Kripke-Kleene semantics of the constructed ADF.*

We now proceed to the proof for $\mathcal{I} = \mathcal{U}$.

in DP: *To decide whether a statement s is not true in the ultimate grounded pair we can directly use Algorithm 1. The correctness of the algorithm is shown in Lemma 4.13. This is a non-deterministic algorithm, witnessing that this decision problem is in* NP. *Therefore deciding whether s is true in the ultimate grounded pair is in* coNP. *For deciding whether s is not false in the ultimate grounded pair a slight adaption of Algorithm 1 (guess (X, Y) with $s \in Y$) is sufficient to show that this is also a problem in* NP, *thus the complementary problem is again in* coNP. *For deciding whether s is undefined in the ultimate grounded pair we again slightly adapt the algorithm, such that in the guessed pair we have $s \in Y \setminus X$ and directly have due to the proof of Lemma 4.13 that the algorithm returns no iff s is undefined in the ultimate grounded pair. Therefore deciding whether a statement is undefined in the grounded pair is a problem in* NP. *Thus combining these checks for all statements in S for the pair to verify we arrive at an algorithm witnessing that this is a problem in* DP.

DP-hard: *For* DP*-hardness, as for the approximate operator, consider the proof of Proposition 4.11. The pair given for verification in that proof is complete iff it is the ultimate grounded pair.* □

We next ask whether there exists a *non-trivial* admissible pair, that is, if at least one statement has a truth value other than unknown. Clearly, we can guess a pair and perform the coNP-check to show that it is admissible. The next result shows that this is also the best we can do.

Theorem 4.15. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Exists}^{\mathcal{I}}_{adm}$ is Σ^P_2-complete.*

Proof. in Σ^P_2: *Let $\mathcal{O} \in \mathcal{I}$. We guess a pair (X, Y) and verify that $X \subseteq Y$ and $(\emptyset, S) <_i (X, Y)$ in polynomial time, and $(X, Y) \leq_i \mathcal{O}(X, Y)$ using the* NP *oracle (Lemma 3.17, Items 1, 3, and 5).*

Σ_2^P-hard: *We provide a reduction from the Σ_2^P-hard problem QBF$_{2,\exists}$-TRUTH. Let $\exists P \forall Q \psi$ be a QBF. Construct $\Xi = \mathsf{RED}_2(P, Q, \psi)$ as defined in Reduction 3.2. Further let $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. Due to Lemma 3.14 there exists a non-trivial admissible pair of $\mathcal{O}$ iff $\exists P \forall Q \psi$ is true.* □

Lemma 3.4 implies the same complexity for the existence of non-trivial complete and preferred pairs.

Corollary 4.16. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$ and $\sigma \in \{com, pre\}$. $\mathsf{Exists}_\sigma^\mathcal{O}$ is Σ_2^P-complete.*

By corollary to Theorem 4.14, the existence of a non-trivial grounded pair can be decided in DP by testing whether the trivial pair $(\emptyset, S)$ is (not) a fixpoint of the relevant operator. The following result shows that this bound can be improved.

Proposition 4.17. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Exists}_{grd}^\mathcal{I}$ is coNP-complete.*

Proof. *Let Ξ be an ADF. Obviously, Ξ has a non-trivial approximate grounded semantics iff the trivial pair $(\emptyset, S)$ is not a fixpoint of $\mathcal{G}_\Xi$, so we show that the co-problem (deciding whether $\mathcal{G}_\Xi(\emptyset, S) = (\emptyset, S)$) is NP-complete.*

in NP: *We have that $\mathcal{G}_\Xi(\emptyset, S) = (\emptyset, S)$ iff $\emptyset \subseteq \mathcal{G}'_\Xi(\emptyset, S) \subseteq \emptyset$ and $S \subseteq \mathcal{G}''_\Xi(\emptyset, S) \subseteq S$. So mainly we have to verify $\mathcal{G}'_\Xi(\emptyset, S) \subseteq \emptyset$ and $S \subseteq \mathcal{G}''_\Xi(\emptyset, S)$. By Lemma 3.17, the first part can be decided in P (item 1) and the second part in NP (item 4).*

NP-hard: *We give a reduction from SAT. Let ψ be a propositional formula over vocabulary P. Define an ADF $D = (S, L, C)$ with $S = P \cup \{z\}$ for $z \notin P$ and $\varphi_p = p$ for $p \in P$ and $\varphi_z = z \wedge \psi$. It is readily verified that by definition every statement has a parent that is undecided in $(\emptyset, S)$ and thus $\mathcal{G}'_\Xi(\emptyset, S) = \emptyset$. Furthermore, $P \subseteq \mathcal{G}'_\Xi(S, \emptyset)$ is easy to show. Thus $S \subseteq \mathcal{G}'_\Xi(S, \emptyset)$ iff $z \in \mathcal{G}'_\Xi(S, \emptyset)$ iff there is an $M \subseteq S$ with $par(z) \setminus M \subseteq S \setminus \emptyset$ and $M \models \varphi_z$ iff there is an $M \subseteq S$ with $M \models \varphi_z$ iff $\varphi_z = z \wedge \psi$ is satisfiable iff ψ is satisfiable.*

For $\mathcal{I} = \mathcal{U}$, the proof is analogous to the one above – we show NP-completeness of the complementary problem.

in NP: *We have to verify $\mathcal{U}'_\Xi(\emptyset, S) \subseteq \emptyset$ and $S \subseteq \mathcal{U}''_\Xi(\emptyset, S)$. By Lemma 3.17(2) and Lemma 3.17(4), this can be done in NP.*

NP-hard: *The construction is the same as for $\mathcal{I} = \mathcal{G}$.* □

Using the result for existence of non-trivial admissible pairs, the verification complexity for the preferred semantics is straightforward to obtain, similarly as in the case of AFs [22].

Proposition 4.18. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Ver}_{pre}^\mathcal{I}(X, Y)$ is Π_2^P-complete.*

Proof. in Π_2^P: *Let Ξ be an ADF and $X \subseteq Y \subseteq S$. To show that (X, Y) is not preferred, we guess a pair (M, N) with $(X, Y) <_i (M, N)$ and use the NP oracle to show that (M, N) is a complete pair (which can be done in DP).*

Π_2^P-hard: *Consider the complementary problem, that is, deciding whether a given pair is not a preferred pair. Even for the special case of the pair $(\emptyset, S)$, Theorem 4.15 shows that this problem is Σ_2^P-hard.* □

We now move on to query-based reasoning. Similarly as before, we mainly utilize the operator for the upper bound to show hardness. Due to this reason, and for the sake of uniformity of proving results for both operators, we slightly deviate from our definition of credulous and skeptical reasoning and show hardness for the question whether a σ-pair exists such that the given statement is false, respectively ask whether the statement is false in all σ-pairs. For the admissibility-based semantics it is straightforward to see that these problems can be reduced to each other. For querying statement s in an ADF Ξ consider the modified ADF $\hat{\Xi}$ with a fresh statement $\hat{s}$ and acceptance condition $\varphi_{\hat{s}} = \neg s$. Let $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$ and $\hat{\mathcal{O}} \in \{\mathcal{G}_{\hat{\Xi}}, \mathcal{U}_{\hat{\Xi}}\}$. It holds that (X, Y) is admissible for $\mathcal{O}$ with $s \in (S \setminus Y)$ iff $(X \cup \{\hat{s}\}, Y)$ is admissible for $\hat{\mathcal{O}}$. Likewise, it holds that for all preferred pairs (X, Y) of $\mathcal{O}$ we have $s \in (S \setminus Y)$ iff in all preferred pairs $(\hat{X}, \hat{Y})$ of $\hat{\mathcal{O}}$ we have $\hat{s} \in \hat{X}$. Further, s is false in the grounded pair of $\mathcal{O}$ iff $\hat{s}$ is true in the grounded pair of $\hat{\mathcal{O}}$.

We now show that on general ADFs credulous reasoning with respect to admissibility is harder than on AFs. By Lemma 3.5, the same lower bound holds for complete and preferred semantics.

Proposition 4.19. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Cred}^{\mathcal{I}}_{adm}$ is Σ^P_2-complete.*

Proof. *Membership is given by Theorem 3.18. Hardness is shown by a reduction from the Σ^P_2-hard problem QBF$_{2,\exists}$-TRUTH. Let $\exists P \forall Q \psi$ be a QBF. Construct $\Xi = \mathsf{RED}_2(P, Q, \psi)$ as defined in Reduction 3.2. Further let $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$. Due to Lemma 3.14 there exists an admissible pair (X, Y) of $\mathcal{O}$ with $z \in (S \setminus Y)$ iff $\exists P \forall Q \psi$ is true.* □

For credulous and skeptical reasoning with respect to the grounded semantics, we first observe that the two coincide since there is always a unique grounded pair. Furthermore, a statement s is true in the approximate grounded pair iff s is true in the least fixpoint (of $\mathcal{G}_\Xi$) iff s is true in all fixpoints iff there is no fixpoint where s is undecided or false. This condition can be encoded in propositional logic and leads to the next result. For the ultimate operator we apply Algorithm 1. For coNP-hardness the proof of [9, Proposition 13] can be easily adapted.

Proposition 4.20. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. Both $\mathsf{Cred}^{\mathcal{I}}_{grd}$ and $\mathsf{Skept}^{\mathcal{I}}_{grd}$ are* coNP*-complete.*

Proof. *For showing the membership result for $\mathcal{I} = \mathcal{U}$, consider verifying that a statement s is not true in the ultimate grounded pair. Algorithm 1 decides this via a guess and check approach. Thus deciding whether a statement is not true can be done in* NP*. The complementary problem decides if a given statement is true in the ultimate grounded pair. This yields* coNP *membership for $\mathsf{Cred}^{\mathcal{U}}_{grd}$. For hardness the proof for both operators is the same. For showing the results for $\mathcal{I} = \mathcal{G}$ consider the following proof.*

in coNP: *We reduce to unsatisfiability checking in propositional logic. Let $\Xi = (S, L, C)$ be an ADF with $S = \{s_1, \ldots, s_n\}$ and assume we want to verify that s_k is true in the grounded pair of Ξ for some $1 \leq k \leq n$. We again extend Reduction 3.3; additionally define the formulas*

$$\phi'_{\mathsf{cfp}} = \phi_{\mathsf{cfp}}[p/p' : p \in P] \qquad \text{(renamed copy of } \phi_{\mathsf{cfp}}\text{)}$$
$$\psi = (\phi_{\mathsf{fp}} \wedge \neg t_k \wedge u_k) \vee (\phi'_{\mathsf{fp}} \wedge \neg u'_k)$$

We claim that ψ is unsatisfiable iff there is no consistent fixpoint where s_k is unknown or false.

1. *$\phi_{\mathsf{cfp}} \wedge \neg t_k \wedge u_k$ is unsatisfiable iff there is no consistent fixpoint where s_k is undefined:*

"if": *Let $\phi_{\mathsf{cfp}} \wedge \neg t_k \wedge u_k$ be satisfiable. Then there is an interpretation $I \subseteq P$ such that $I \models \phi_{\mathsf{cfp}}$ and $I \models \neg t_k \wedge u_k$. Using Lemma 3.15 we can construct a consistent pair (T, U) and show that it is a fixpoint of $\mathcal{G}_\Xi$ with $s_k \in U \setminus T$.*

"only if": *Let $T \subseteq U \subseteq S$ such that $\mathcal{G}_\Xi(T, U) = (T, U)$ and $s_k \in U \setminus T$. Lemma 3.15 yields an interpretation $I \subseteq P$ such that $I \models \phi_{\mathsf{cfp}}$ and $I \models \neg t_k \wedge u_k$.*

2. *$\phi'_{\mathsf{cfp}} \wedge \neg u'_k$ is unsatisfiable iff there is no consistent fixpoint where s_k is false: similar.*

coNP-hard: *Let ψ be a propositional formula over vocabulary P. Define the ADF $D = (S, L, C)$ with $S = P \cup \{z\}$, $\varphi_p = \neg\psi$ for $p \in P$, and $\varphi_z = \bigwedge_{p \in P} \neg p$. We show that z is true in the grounded semantics of $\mathcal{G}_D$ iff ψ is a tautology.*

"if": *Let ψ be a tautology. Then $\neg\psi$ is unsatisfiable and $p \notin \mathcal{G}''_D(\emptyset, S)$ for all $p \in P$. Obviously φ_z is satisfiable whence $z \in \mathcal{G}''_D(\emptyset, S)$. Thus $\mathcal{G}''_D(\emptyset, S) = \{z\}$. Furthermore $\mathcal{U}'_D(\emptyset, S) = \emptyset$, since no acceptance condition is a tautology. Therefore also $\mathcal{G}'_D(\emptyset, S) = \emptyset$. Thus $\mathcal{G}_D(\emptyset, S) = (\emptyset, \{z\})$. Now since z does not occur in the acceptance formula of z, it is clear that $\mathcal{G}_D(\emptyset, \{z\}) = (\{z\}, \{z\}) = \mathcal{G}_D(\{z\}, \{z\})$. Thus z is true in the grounded semantics of $\mathcal{G}_D$.*

"only if": *Let $lfp(\mathcal{G}_D) = (X, Y)$ and $z \in X$. By the acceptance condition of z and the fact that (X, Y) is a fixpoint of $\mathcal{G}_D$ we get $P \cap Y = \emptyset$. Since $X \subseteq Y$ we have $(X, Y) = (\{z\}, \{z\})$. Assume to the contrary that ψ is not a tautology. Then $\neg\psi$ is satisfiable and $P \subseteq Y = \mathcal{G}''_D(\emptyset, S)$. Contradiction.* □

Regarding skeptical reasoning for the remaining semantics we need only show the results for complete and preferred semantics, in all other cases the complexity coincides with credulous reasoning or is trivial. For complete semantics it is easy to see that skeptical reasoning coincides with skeptical reasoning under grounded semantics, since the grounded pair is the $\leq_i$-least complete pair.

Corollary 4.21. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$.* $\mathsf{Skept}^{\mathcal{I}}_{com}$ *is* coNP*-complete.*

Similar to reasoning on AFs, we step up one level of the polynomial hierarchy by changing from credulous to skeptical reasoning with respect to preferred semantics, which makes skeptical reasoning under preferred semantics particularly hard. We apply proof ideas by [27] to prove Π_3^P-hardness.

Theorem 4.22. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$.* $\mathsf{Skept}^{\mathcal{I}}_{pre}$ *is Π_3^P-complete.*

Proof. Membership is given by Theorem 3.18. Hardness is shown by a reduction from the Π_3^P-hard problem QBF$_{3,\forall}$-TRUTH. Let $\forall P \exists Q \forall R \psi$ be a QBF. We define an ADF Ξ as follows:

- *$S = P \cup Q \cup \text{-}Q \cup R \cup \{f\}$, with $\text{-}Q = \{\text{-}q \mid q \in Q\}$,*
- *$\varphi_p = p$ for $p \in P$,*
- *$\varphi_q = \neg f \wedge \neg\text{-}q$ for $q \in Q$,*
- *$\varphi_{\text{-}q} = \neg f \wedge \neg q$ for $\text{-}q \in \text{-}Q$,*
- *$\varphi_r = \neg r$ for $r \in R$,*
- *$\varphi_f = \neg f \wedge \neg\psi$.*

Let $\mathcal{O} \in \{\mathcal{G}_{\Xi}, \mathcal{U}_{\Xi}\}$. We now show that all preferred pairs (X, Y) of $\mathcal{O}$ have $f \in S \setminus Y$ iff $\forall P \exists Q \forall R \psi$ is true. First observe some helpful facts. For each $P' \subseteq P$ there is a preferred pair (X, Y) of $\mathcal{O}$ with $P' \subseteq X$ and $(P \setminus P') \subseteq (S \setminus Y)$, i.e. for each two-valued valuation on P there exists a preferred pair assigning to the statements in P exactly these values. In addition for each preferred pair (X, Y) of $\mathcal{O}$ it holds that $P \cap (Y \setminus X) = \emptyset$. These two facts are shown in Lemma 3.12. Further for all admissible pairs (X, Y) of $\mathcal{O}$ we have that $f \notin X$ due to Lemma 3.10 and if $f \in (Y \setminus X)$ then also $(Q \cup \text{-}Q) \subseteq (Y \setminus X)$ due to Lemma 3.11.

"if": *Assume that the formula is valid. Let $P' \subseteq P$. For such a P' we know that there is a $Q' \subseteq Q$ such that for any $R' \subseteq R$ we have $P' \cup Q' \cup R' \models \psi$. We now show there is a preferred pair (X', Y') with $X' = P' \cup Q' \cup \text{-}(Q \setminus Q')$ and $Y' = X' \cup R$ for $\mathcal{O}$. We set $\text{-}(Q \setminus Q') = \{\text{-}q \mid q \in (Q \setminus Q')\}$. It is easy to see that $P' \subseteq \mathcal{G}'_{\Xi}(X', Y') \subseteq \mathcal{U}'_{\Xi}(X', Y')$. Likewise since $f \in S \setminus Y'$ also $Q' \subseteq \mathcal{G}'_{\Xi}(X', Y') \subseteq \mathcal{U}'_{\Xi}(X', Y')$. Similarly it follows that $\text{-}(Q \setminus Q') \subseteq \mathcal{G}'_{\Xi}(X', Y') \subseteq \mathcal{U}'_{\Xi}(X', Y')$. Proving that $(S \setminus Y') \setminus \{f\}$ is not contained in $\mathcal{U}''_{\Xi}(X', Y')$ proceeds analogous. The statements $r \in R$ are always undecided in admissible pairs for both operators. For proving that (X', Y') is preferred for $\mathcal{O}$, it remains to be shown that $f \notin \mathcal{G}''_{\Xi}(X', Y')$. For all Z with $X' \subseteq Z \subseteq Y'$ we have that $Z \models \psi$, since $Z \cap P = P'$, $Z \cap Q = Q'$ and by assumption we know that $P' \cup Q' \cup R' \models \psi$. Therefore $Z \not\models \varphi_f$ and thus $f \notin \mathcal{U}_{\Xi}(X', Y')$ and (X', Y') is admissible for $\mathcal{O}$. Since only R are undecided in this pair and these statements are undecided in all admissible pairs, we can conclude that (X', Y') is indeed a preferred pair w.r.t. $\mathcal{O}$.*

Now we know that there exists a preferred pair which sets P' to true and f to false. To see that there is no preferred pair (X'', Y'') which assigns the same values to the statements in P and $f \in Y''$, it suffices to show that for this case then also $Q, \text{-}Q \subseteq Y'' \setminus X''$, which holds due to Lemma 3.11. More formally, assume (X'', Y'') is admissible for $\mathcal{O}$ and $P' \subseteq X''$, $(P \setminus P') \subseteq (S \setminus Y)$ and $f \in (Y'' \setminus X'')$. We can conclude that $Q, \text{-}Q \subseteq Y'' \setminus X''$ due to Lemma 3.11 and thus $X'' = P'$ and $S \setminus Y'' = (P \setminus P')$. Hence $(X'', Y'') <_i (X', Y')$ and the former pair cannot be preferred for $\mathcal{O}$. Summarizing, all preferred pairs set a $P' \subseteq P$ to true and $P \setminus P'$ to false and for each such choice there exists a preferred pair setting f to false. Further if for such a choice a preferred pair exists with f set to false, we know that there is no preferred pair with the same assignment to the statements in P and setting f not to false. Thus any preferred pair of $\mathcal{O}$ sets f to false.

"only if": *Assume that in any preferred pair of $\mathcal{O}$ we have that $f \in S \setminus Y$. As in the "if" direction we know that for any $P' \subseteq P$ there exists a preferred pair (X, Y) with $P \cap (Y \setminus X) = \emptyset$ and $P' \subseteq X$. Since $f \in S \setminus Y$ we have that in any such preferred pair also $(Q \cup \text{-}Q) \cap (Y \setminus X) = \emptyset$, since otherwise it would not be maximal w.r.t. $\leq_i$ by a similar argument as in the proof of Lemma 3.12. Further we know that $R \subseteq Y \setminus X$. Let $Q' = Q \cap X$. Thus for any $R' \subseteq R$ we have $P' \cup Q' \cup R' \not\models \varphi_f$, since $f \in S \setminus Y$ and hence $P' \cup Q' \cup R' \models \psi$. Since for any $P' \subseteq P$ such a preferred pair exists we can conclude that the QBF is valid.* □

4.3. Two-valued semantics

The complexity results we have obtained so far might lead the reader to ask why we bother with the approximate operator $\mathcal{G}_{\Xi}$ at all: the ultimate operator $\mathcal{U}_{\Xi}$ is at least as precise and for all admissibility-based and most conflict-free-based semantics considered up to now, it has the same computational costs. We now show that for the verification of two-valued stable models, the operator for the upper bound plays no role and therefore the complexity difference between the lower bound operators for approximate (in P) and ultimate (coNP-hard) semantics comes to bear.

For the ultimate two-valued stable semantics, Brewka et al. [11] already have some complexity results: model verification is in DP (Proposition 8), and model existence is Σ_2^P-complete (Theorem 9). We will show next that we can do better for the approximate version.

Proposition 4.23. *Let Ξ be an ADF and $X \subseteq Y \subseteq S$. Checking that X is the least fixpoint of $\mathcal{G}'_\Xi(\cdot, Y)$ can be done in polynomial time.*

Proof. *We provide the following polynomial-time decision procedure with input Ξ, X, Y.*

1. *Set $i = 0$ and $X_0 = \emptyset$.*
2. *For each statement $s \in S$, do the following:*
   (a) *If $par(s) \cap (Y \setminus X_i) = \emptyset$ and $C_s(par(s) \cap X_i) = \mathbf{t}$, then set $s \in X_{i+1}$.*
3. *If $X_{i+1} = X_i = X$ then return "Yes".*
4. *If $X_{i+1} = X_i \subsetneq X$ then return "No".*
5. *If $X_{i+1} \not\subseteq X$ then return "No".*
6. *Increment i and go to step 2.*

Overall, the loop between steps 2 and 6 is executed at most $|S|$ times, since $X_i \subseteq X_{i+1}$ for all $i \in \mathbb{N}$ and we can add at most all statements one by one. In each execution of the loop, step 2a is executed $|S|$ times. The conditions of step 2a, in particular $par(s) \cap X_i \models \varphi_s$, can be verified in polynomial time.

It remains to show that X is the least fixpoint of $\mathcal{G}'_\Xi(\cdot, Y)$ iff the procedure returns "Yes".

"if": *Assume the procedure returned "Yes" on input Ξ, X, Y.*

- *X is a fixpoint of $\mathcal{G}'_\Xi(\cdot, Y)$, that is, $\mathcal{G}'_\Xi(X, Y) = X$:*

  "$\subseteq$": *Let $s \in \mathcal{G}'_\Xi(X, Y)$. Then there is a $B \subseteq X \cap par(s)$ such that $C_s(B) = \mathbf{t}$ and $par(s) \setminus B \subseteq S \setminus Y$. As in the proof of Proposition 3.16, we get that $B = X \cap par(s)$, $C_s(par(s) \cap X) = \mathbf{t}$ and $par(s) \cap (Y \setminus X) = \emptyset$. Since the procedure answered "Yes", there was an $i \in \mathbb{N}$ with $X_{i+1} = X_i = X$. From step 2a of the procedure, we know that $par(s) \cap (Y \setminus X_i) = \emptyset$ and $C_s(par(s) \cap X_i) = \mathbf{t}$ means that $s \in X_{i+1} = X$.*

  "$\supseteq$": *Let $s \in X$. Since the procedure answered "Yes", there was an $i \in \mathbb{N}$ with $X_{i+1} = X_i = X$. Now $s \in X_{i+1}$ by step 2a of the procedure means that $par(s) \cap (Y \setminus X_i) = \emptyset$ and $C_s(par(s) \cap X_i) = \mathbf{t}$. Thus there exists a $B = par(s) \cap X$ with $C_s(B) = \mathbf{t}$ and $par(s) \setminus B \subseteq S \setminus Y$, and $s \in \mathcal{G}'_\Xi(X, Y)$.*

- *X is the least fixpoint: Assume to the contrary that there is some $X' \subsetneq X$ that is a fixpoint of $\mathcal{G}'_\Xi(\cdot, Y)$. But then step 4 of the procedure would have detected $X_{i+1} = X_i = X' \subsetneq X$ and returned "No", contradiction.*

"only if": *Let X be the least fixpoint of $\mathcal{G}'_\Xi(\cdot, Y)$ and assume to the contrary that the procedure answered "No".*

- *The procedure answered "No" in step 4. By the argument above, we can show that there is a fixpoint $X' \subsetneq X$, contradiction.*
- *The procedure answered "No" in step 5. We have $X_{i+1} \not\subseteq X$ for some $i \in \mathbb{N}$, that is, there is some $s \in X_{i+1}$ with $s \notin X$. Since $s \in X_{i+1}$, we have $par(s) \cap (Y \setminus X_i) = \emptyset$ and $C_s(par(s) \cap X_i) = \mathbf{t}$. Since the procedure did not terminate with X_i already, we know that $X_i \subseteq X$. Therefore, $par(s) \cap (Y \setminus X) = \emptyset$ and $C_s(par(s) \cap X) = \mathbf{t}$. This means $s \in \mathcal{G}'_\Xi(X, Y) = X$. Contradiction.* ∎

In particular, the procedure can decide whether Y is the least fixpoint of $\mathcal{G}'_{\Xi}(\cdot, Y)$, that is, whether (Y, Y) is a two-valued stable model of $\mathcal{G}_{\Xi}$. This yields the next result.

Theorem 4.24. 1. $\mathsf{Ver}^{\mathcal{G}}_{2st}$ *is in* P, *and*

2. $\mathsf{Exists}^{\mathcal{G}}_{2st}$ *is* NP*-complete.*

Proof. Let Ξ be an ADF and $X \subseteq S$.

1. *We have to verify that X is the least fixpoint of the operator $\mathcal{G}'_{\Xi}(\cdot, X)$, which can be done in polynomial time by Proposition 4.23.*
2. *Deciding whether Ξ has a two-valued stable model is* NP*-complete:*

   in NP: *To decide whether there is a two-valued stable model, we guess a set $X \subseteq S$ and verify as above that (X, X) is indeed a two-valued stable model.*

   NP-hard: *Carries over from AFs.* □

The hardness direction of the second part is clear since the respective result from stable semantics of abstract argumentation frameworks carries over.

Brewka et al. [11] showed that $\mathsf{Ver}^{\mathcal{U}}_{2st}$ is in DP (Proposition 8). We can improve that upper bound to coNP: basically the operator for the upper bound (contributing the NP part) is not really needed. We furthermore also provide a hardness proof for coNP.

Proposition 4.25. $\mathsf{Ver}^{\mathcal{U}}_{2st}$ *is* coNP*-complete.*

Proof. in coNP: *Given an ADF $\Xi = (S, L, C)$ and a set $M \subseteq S$ we first construct the reduct Ξ^M in polynomial time. Now M is an ultimate two-valued stable model of Ξ iff all statements in M are true in the grounded semantics of Ξ^M and (M, M) is a model of Ξ. Verifying if a statement is true in the ultimate grounded pair of an ADF is* coNP*-complete due to Proposition 4.20. Thus verifying that all statements in M are true in the ultimate grounded pair is likewise a problem in* coNP*. Verifying if (M, M) is a model of Ξ can be achieved in polynomial time. This means that* $\mathsf{Ver}^{\mathcal{U}}_{2st}$ *is in* coNP.

coNP-hard: *Let ψ be a propositional formula over a vocabulary P. We define an ADF D over statements P with $\varphi_p = \psi$ for all $p \in P$. When we apply $\mathcal{U}'_D$ to the pair $(\emptyset, P)$, there are only two possible outcomes: either $\psi = \psi^{(\emptyset,P)} = \varphi_p^{(\emptyset,P)}$ is a tautology, then $p \in \mathcal{U}'_D(\emptyset, P)$ for all $p \in P$, that is $\mathcal{U}'_D(\emptyset, P) = P$; otherwise ψ is refutable and accordingly $\mathcal{U}'_D(\emptyset, P) = \emptyset$. Furthermore, in the former case it follows from $\leq_i$-monotonicity of $\mathcal{U}_D$ that $P = \mathcal{U}'_D(\emptyset, P) \subseteq \mathcal{U}'_D(P, P)$. Thus ψ is a tautology if and only if P is a fixpoint of $\mathcal{U}'_D(\cdot, P)$ and $\emptyset$ is not. Now*

$$
\begin{aligned}
&P \text{ is an ultimate two-valued stable model of } D\\
\text{iff } &P \text{ is the least fixpoint of } \mathcal{U}'_D(\cdot, P)\\
\text{iff } &\mathcal{U}'_D(\emptyset, P) = P = \mathcal{U}'_D(P, P)\\
\text{iff } &\psi \text{ is a tautology}
\end{aligned}
$$

□

We now turn to the credulous and skeptical reasoning problems for the two-valued semantics. We first recall that a two-valued pair (X, X) is a supported model (or model for short) of an ADF Ξ iff $\mathcal{G}_{\Xi}(X, X) = (X, X)$. Thus it could equally well be characterized by the two-valued operator by saying that X is a model iff $G_{\Xi}(X) = X$. Now since $\mathcal{U}_{\Xi}$ is the ultimate approximation of G_{Ξ}, also $\mathcal{U}_{\Xi}(X, X) = (X, X)$ in this case. Rounding up, this recalls that approximate and ultimate two-valued supported models coincide. Hence we get the following results for reasoning with this semantics.

Corollary 4.26. *Consider any $\mathcal{I} \in \{\mathcal{G}, \mathcal{U}\}$. $\mathsf{Cred}^{\mathcal{I}}_{2su}$ is NP-complete and $\mathsf{Skept}^{\mathcal{I}}_{2su}$ is coNP-complete.*

Proof. The membership parts are clear since $\mathsf{Ver}^{\mathcal{I}}_{2su}$ is in P. Hardness carries over from AFs [22]. □

For the approximate two-valued stable semantics, the fact that model verification can be decided in polynomial time leads to the next result.

Corollary 4.27. *$\mathsf{Cred}^{\mathcal{G}}_{2st}$ is NP-complete and $\mathsf{Skept}^{\mathcal{G}}_{2st}$ is coNP-complete.*

Proof. The membership parts are clear since $\mathsf{Ver}^{\mathcal{G}}_{2st}$ is in P. Hardness carries over from AFs [22]. □

For the ultimate two-valued stable semantics, things are bit more complex. The following result was already presented by Brewka et al. [11], however they had to leave out the proof due to space restrictions. We present the proof (inspired by the proof of [20, Theorem 6.12]) here for completeness and since we will need it later on.

Theorem 4.28 ([11, Theorem 9]). *Deciding whether a given ADF has an ultimate two-valued stable model is Σ^{P}_2-complete.*

Proof. Let $\Xi = (S, L, C)$ be an ADF. For membership, we first guess a set $M \subseteq S$. We can verify in polynomial time that M is a two-valued supported model of Ξ, and compute the reduct Ξ_M. Using the NP oracle, we can compute the grounded semantics (K', K'') of the reduct in polynomial time. It then only remains to check $K' = M$.

For hardness, we provide a reduction from the Σ^P_2-complete problem of deciding whether a $QBF_{2,\exists}$-formula is valid. Let $\exists P \forall Q \psi$ be an instance of $QBF_{2,\exists}$-TRUTH where ψ is in DNF and $P, Q \neq \emptyset$. We have to construct an ADF D such that D has a stable model iff $\exists P \forall Q \psi$ is true.

First of all, define $\text{-}P = \{\text{-}p \mid p \in P\}$ for abbreviating the negations of $p \in P$. For guessing an interpretation for P, define the acceptance formulas $\varphi_p = \neg\text{-}p$ and $\varphi_{\text{-}p} = \neg p$ for $p \in P$. Define ψ' as the formula $\psi[\neg p/\text{-}p]$ where all occurrences of $\neg p$ have been replaced by $\text{-}p$. Further add a statement z with $\varphi_z = \neg z \wedge \neg \psi'$, an integrity constraint that ensures truth of ψ' in any model. For $q \in Q$ we set $\varphi_q = \psi'$. Thus we get the statements $S = P \cup \text{-}P \cup Q \cup \{z\}$. We have to show that D has a stable model iff $\exists P \forall Q \psi$ is true.

"if": *Let $M_P \subseteq P$ be such that the following formula over vocabulary Q is a tautology:*

$$\phi = \psi^{(M_P, M_P \cup Q)}$$

We now construct a stable model $M = M_P \cup Q \cup \{\text{-}p \in \text{-}P \mid p \notin M_P\}$. We first show that M is a model of D: For each $p \in M_P$, we have $\text{-}p \notin M$ by definition and hence $M \models \varphi_p = \neg\text{-}p$. Conversely, if $p \notin M_P$ then $\text{-}p \in M$ and $M \models \varphi_{\text{-}p} = \neg p$. For $q \in Q$, we have that $\varphi_q = \psi'$ and so we have to show $M \models \psi'$. This is however immediate since ϕ (the partial evaluation of ψ with M as interpretation for P) is a tautology. Finally, by definition $z \notin M$, and since $M \models \psi'$ we get $M \not\models \varphi_z = \neg z \wedge \neg \psi'$ as required.

To show that M is a stable model, we have to show that all statements in M are true in the ultimate Kripke-Kleene semantics of the reduct D_M. The reduct is given by

- $D_M = (M, L_M, C_M)$ *with*
- $\varphi_p = \neg\mathbf{f}$ *for $p \in M$,*
- $\varphi_{\text{-}p} = \neg\mathbf{f}$ *for $\text{-}p \in M$,*
- $\varphi_q = \psi'^{(\emptyset, M)}$.

The computation of the Kripke-Kleene semantics starts with $(\emptyset, M)$ *and leads to the first revision* $(K_0', K_0'') = \mathcal{U}_{\Xi}(\emptyset, M)$. *Since the acceptance condition of any* $p, \text{-}p \in M$ *is tautological, we have* $p, \text{-}p \in K_0'$, *that is, the statements* $p, \text{-}p \in M$ *are considered true. For the next step, the acceptance formula of any* $q \in Q$ *can thus be simplified to*

$$\begin{aligned}\varphi_q^{(M\setminus Q, M)} &= \left(\psi'^{(\emptyset,M)}\right)^{(M\setminus Q,M)} \\ &= \psi'^{(M\setminus Q,M)} \\ &= \psi'[p/\mathbf{f} : p \notin M, \text{-}p/\mathbf{f} : \text{-}p \notin M, p/\mathbf{t} : p \in M, \text{-}p/\mathbf{t} : \text{-}p \in M],\end{aligned}$$

a formula over Q *that is equivalent to* $\phi = \psi^{(M_P, M_P \cup Q)}$. *By presumption,* ϕ *is a tautology. Hence at this point all acceptance formulas partially evaluated by* (K_0', K_0'') *are tautologies and thus* $\mathcal{U}_{\Xi}(K_0', K_0'') = (M, M)$, *which has already been shown to be a fixpoint of* $\mathcal{U}_{\Xi}$.

"only if": *Let* $M \subseteq S$ *be an ultimate two-valued stable model of* D. *We have to show that* $\exists P \forall Q \psi$ *is true. Define* $M_P = M \cap P$ *and* $\phi = \psi^{(M_P, M_P \cup Q)}$. *We show that* ϕ *is a tautology.*

First of all, since M *is a model of* D_M *we have* $z \notin M$: *assume to the contrary that* $z \in M$, *then* M *is a model for* $\varphi_z = \neg z \wedge \neg \psi' \equiv \mathbf{f} \wedge \neg \psi'$, *contradiction. Hence* $M \not\models \neg z \wedge \neg \psi'$, *that is,* $M \not\models \neg \psi'$. *This shows that* $M \models \psi'$, *that is,* $M \models \varphi_q$ *for all* $q \in Q$, *whence* $Q \subseteq M$. *Thus the evaluation of* $p \in P$ *and* $\text{-}p \in \text{-}P$ *defined by* M *shows the truth of the formula*

$$\psi'^{(M,M)} = \psi'[p/\mathbf{t} : p \in M, \text{-}p/\mathbf{t} : \text{-}p \in M, p/\mathbf{f} : p \notin M, \text{-}p/\mathbf{f} : \text{-}p \notin M][q/\mathbf{t} : q \in Q]$$

Now since M *is a stable model of* D, *the pair* (M, M) *is the ultimate grounded semantics of the reduct* D_M *again given by*

- $D_M = (M, L_M, C_M)$ *with*
- $\varphi_p = \neg \mathbf{f}$ *for* $p \in M$,
- $\varphi_{\text{-}p} = \neg \mathbf{f}$ *for* $\text{-}p \in M$,
- $\varphi_q = \psi'^{(\emptyset,M)}$.

To show that ϕ *is a tautology, assume to the contrary that* ϕ *is refutable. As observed in the "if" part,* ϕ *is equivalent to the formula* $\varphi_q^{(M\setminus Q,M)}$. *Thus also* φ_q *is refutable, whence* $q \notin \mathcal{U}_{D_M}'(\emptyset, M)$ *for all* $q \in Q$ *and* $\mathcal{U}_{D_M}'(\emptyset, M) = M \setminus Q$. *Furthermore we know that* $\mathcal{U}_{D_M}''(\emptyset, M) = M$. *Now* $\varphi_q^{(M\setminus Q,M)}$ *is refutable and thus* $\mathcal{U}_{D_M}(M \setminus Q, M) = (M \setminus Q, M)$. *Since* $Q \neq \emptyset$, *we find that* (M, M) *is not the least fixpoint of* $\mathcal{U}_{D_M}$. *Contradiction.* □

The hardness reduction in this proof makes use of a statement z that is false in any ultimate two-valued stable model. This can be used to show the same hardness for the credulous reasoning problem for this semantics: we introduce a new statement x that behaves just like $\neg z$, then x is true in some model if and only if there exists a model.

Proposition 4.29. *The problem* $\mathsf{Cred}_{2st}^{\mathcal{U}}$ *is* Σ_2^P*-complete.*

Proof. in Σ_2^P: *Let* Ξ *be an ADF and* $s \in S$. *We can guess a set* $X \subseteq S$ *with* $s \in X$ *and verify in* coNP *that it is an ultimate two-valued stable model.*

Σ_2^P-hard: *Let* $\exists P \forall Q \psi$ *be a QBF. We use the same ADF construction as in the hardness proof of* $\mathsf{Exists}_{2st}^{\mathcal{U}}$ *and augment* D *by an additional statement* x *with* $\varphi_x = \neg z$. *It is clear that in any model of* D, z *must be false and so* x *must be true. So* x *is true in some two-valued stable model of* D *iff* D *has a two-valued stable model iff* $\exists P \forall Q \psi$ *is true.* □

A similar argument works for the skeptical reasoning problem: Given a QBF $\forall P \exists Q \psi$, we construct its negation $\exists P \forall Q \neg \psi$, whose associated ADF D has an ultimate two-valued stable model (where z is false) iff $\exists P \forall Q \neg \psi$ is true iff the original QBF $\forall P \exists Q \psi$ is false. Hence $\forall P \exists Q \psi$ is true iff z is true in all ultimate two-valued stable models of D.

Proposition 4.30. *The problem* $\mathsf{Skept}^{\mathcal{U}}_{2st}$ *is* Π^P_2*-complete.*

Proof. in Π^P_2: *Let* Ξ *be an ADF and* $s \in S$. *To decide the co-problem, we guess a set* $X \subseteq S$ *with* $s \notin X$ *and verify in* coNP *that it is an ultimate two-valued stable model.*

Π^P_2-hard: *Let* $\forall P \exists Q \psi$ *be a QBF with* ψ *in CNF. Define the QBF* $\exists P \forall Q \neg \psi$ *and observe that* $\neg \psi$ *can be transformed into DNF in linear time. We use this new QBF to construct an ADF* D *as we did in the hardness proof of* $\mathsf{Exists}^{\mathcal{U}}_{2st}$. *As observed in the proof of Proposition 4.29, the special statement* z *is false in all ultimate two-valued stable models of* D. *To show that* $\forall P \exists Q \psi$ *is true iff* z *is true in all ultimate two-valued stable models of* D, *we show that* $\forall P \exists Q \psi$ *is false iff* D *has an ultimate two-valued stable model where* z *is false:* $\forall P \exists Q \psi$ *is false iff* $\neg \forall P \exists Q \psi$ *is true iff* $\exists P \forall Q \neg \psi$ *is true iff* D *has an ultimate two-valued stable model iff* D *has an ultimate two-valued stable model where* z *is false.* □

approximate ($\mathcal{G}$), σ	conflict-free	naive	admissible	complete	preferred	grounded	model	stable model
$\mathsf{Ver}^{\mathcal{G}}_{\sigma}$	NP-c (Proposition 4.1)	DP-c (Proposition 4.6)	coNP-c (Proposition 4.10)	DP-c (Proposition 4.11)	Π^P_2-c (Proposition 4.18)	DP-c (Theorem 4.14)	in P ([11, Prop. 5])	in P (Theorem 4.24)
$\mathsf{Exists}^{\mathcal{G}}_{\sigma}$	in P (Proposition 4.8)	in P (Corollary 4.9)	Σ^P_2-c (Theorem 4.15)	Σ^P_2-c (Corollary 4.16)	Σ^P_2-c (Corollary 4.16)	coNP-c (Proposition 4.17)	NP-c ([11, Prop. 5])	NP-c (Theorem 4.24)
$\mathsf{Cred}^{\mathcal{G}}_{\sigma}$	NP-c (Proposition 4.4)	NP-c (Corollary 4.5)	Σ^P_2-c (Proposition 4.19)	Σ^P_2-c (Proposition 4.19, Lemma 3.5)	Σ^P_2-c (Proposition 4.19, Lemma 3.5)	coNP-c (Proposition 4.20)	NP-c (Corollary 4.26)	NP-c (Corollary 4.27)
$\mathsf{Skept}^{\mathcal{G}}_{\sigma}$	trivial	Π^P_2-c (Proposition 4.7)	trivial	coNP-c (Corollary 4.21)	Π^P_3-c (Theorem 4.22)	coNP-c (Proposition 4.20)	coNP-c (Corollary 4.26)	coNP-c (Corollary 4.27)

ultimate ($\mathcal{U}$), σ	conflict-free	naive	admissible	complete	preferred	grounded	model	stable model
$\mathsf{Ver}^{\mathcal{U}}_{\sigma}$	NP-c (Proposition 4.1)	DP-c (Proposition 4.6)	coNP-c ([11, Prop. 10])	DP-c ([11, Cor. 7])	Π^P_2-c (Proposition 4.18)	DP-c ([11, Thm. 6])	in P ([11, Prop. 5])	coNP-c (Proposition 4.25)
$\mathsf{Exists}^{\mathcal{U}}_{\sigma}$	NP-c (Proposition 4.2)	NP-c (Corollary 4.3)	Σ^P_2-c (Theorem 4.15)	Σ^P_2-c (Corollary 4.16)	Σ^P_2-c (Corollary 4.16)	coNP-c (Proposition 4.17)	NP-c ([11, Prop. 5])	Σ^P_2-c (Theorem 4.28)
$\mathsf{Cred}^{\mathcal{U}}_{\sigma}$	NP-c (Proposition 4.4)	NP-c (Corollary 4.5)	Σ^P_2-c (Proposition 4.19)	Σ^P_2-c (Proposition 4.19, Lemma 3.5)	Σ^P_2-c (Proposition 4.19, Lemma 3.5)	coNP-c (Proposition 4.20)	NP-c (Corollary 4.26)	Σ^P_2-c (Proposition 4.29)
$\mathsf{Skept}^{\mathcal{U}}_{\sigma}$	trivial	Π^P_2-c (Proposition 4.7)	trivial	coNP-c (Corollary 4.21)	Π^P_3-c (Theorem 4.22)	coNP-c (Proposition 4.20)	coNP-c (Corollary 4.26)	Π^P_2-c (Proposition 4.30)

Table 2: Complexity results for semantics of Abstract Dialectical Frameworks.

5. Complexity of Bipolar ADFs

In this section, we take a closer look at the special class of ADFs where all links are supporting or attacking, and more importantly the specific link type is known for each link. We first note that since BADFs are a subclass of ADFs, all membership results from the previous sections immediately carry over. However, we can show that many problems will in fact become easier. Intuitively, computing the revision operators is now P-easy because the associated satisfiability/tautology problems only have to treat restricted acceptance formulas. In bipolar ADFs, by definition, if in some three-valued pair (X, Y) a statement s is accepted by a revision operator ($s \in \mathcal{O}'(X, Y)$), it will stay so if we set its undecided supporters to false and its undecided attackers to true. Symmetrically, if a statement is rejected by an operator ($s \notin \mathcal{O}''(X, Y)$), it will stay so if we set its undecided supporters to true and its undecided attackers to false. Hence to decide whether $s \in \mathcal{O}'(X, Y)$ or $s \notin \mathcal{O}''(X, Y)$ for given operator $\mathcal{O}$, pair (X, Y) and statement s, we need only look at *one single interpretation* that can be constructed from the known link types. This is the key idea underlying the next result. Recall that $\mathcal{BG}$ and $\mathcal{BU}$ are the restrictions of the sets of operators $\mathcal{G}$ and $\mathcal{U}$ respectively to BADFs where the type of each link is known.

Proposition 5.1. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$.*

1. $\mathsf{Elem}^{\mathcal{I}'}$ *is in* P.
2. $\mathsf{Elem}^{\mathcal{I}''}$ *is in* P.

Proof. *Let Ξ be a BADF with $L = L^+ \cup L^-$, $\mathcal{O} \in \{\mathcal{G}_\Xi, \mathcal{U}_\Xi\}$, $s \in S$ and $X \subseteq Y \subseteq S$. It suffices to show the claims for $\mathcal{I} = \mathcal{BU}$, since the result that $s \in \mathcal{U}''_\Xi(X, Y)$ is computable in polynomial time implies that deciding $s \in \mathcal{G}''_\Xi(X, Y)$ can likewise be achieved in polynomial time, due to coincidence of the two operators. Further due to Proposition 3.16 we know that deciding $s \in \mathcal{G}'_\Xi(X, Y)$ is a problem in P.*

Recall that for $M \subseteq S$, if a link (z, s) is attacking, then it cannot be the case that $M \not\models \varphi_s$ and $M \cup \{z\} \models \varphi_s$. Similarly if (z, s) is supporting, then it cannot be the case that $M \models \varphi_s$ and $M \cup \{z\} \not\models \varphi_s$. If (x, s) is attacking and supporting then for any $M \subseteq S$ we have $M \models \varphi_s$ iff $M \cup \{z\} \models \varphi_s$, i.e. a change of the truth value of z does not change the evaluation of φ_s.

Given a consistent pair (X, Y) and $s \in S$ we can use a "canonical" interpretation representing all $X \subseteq Z \subseteq Y$ as follows. Note that the aforementioned "redundant" links, i.e. links in the intersection $L^+ \cap L^-$ can be disregarded completely and for ease of notation we will assume in the proof that no such link is present (formally if (x, s) is a redundant link, then we can replace each x in φ_s uniformly with $\mathbf{t}$ or $\mathbf{f}$). Let $Z \subseteq S$, $Z' \subseteq att_\Xi(s)$ and $Z'' \subseteq supp_\Xi(s)$. Then

$$\begin{aligned} & s \in \mathcal{U}'_\Xi(Z, Z) \\ \text{iff } & s \in \mathcal{U}'_\Xi(Z \setminus Z', Z) \\ \text{iff } & s \in \mathcal{U}'_\Xi(Z \setminus Z', Z \cup Z''). \end{aligned}$$

The "if" direction is both times trivially satisfied. This can be seen by the easy fact that if $\varphi_s^{(L,U)}$ is tautological, then so is $\varphi_s^{(L',U')}$ with $(L, U) \leq_i (L', U')$. Suppose the first "only if" does not hold, i.e. the first line holds, but the second is not true. Then there exists a set H with $(Z \setminus Z') \subseteq H \subseteq Z$ such that $H \not\models \varphi_s$. By assumption $Z \models \varphi_s$ and since $H \cup (Z' \cap Z) = Z$ also $H \cup (Z' \cap Z) \models \varphi_s$, which is a contradiction, since $Z' \subseteq att_\Xi(s)$ and thus $(Z' \cap Z) \subseteq att_\Xi(s)$, which implies that there exists a statement in $att_\Xi(s)$ which is not attacking.

Suppose the second only if does not hold, then there exists an H with $(Z \setminus Z') \subseteq H \subseteq (Z \cup Z'')$ such that $H \not\models \varphi_s$. Since we have that $(Z \setminus Z') \subseteq (H \setminus (Z'' \setminus Z)) \subseteq Z$ it follows that $H \setminus (Z'' \setminus Z) \models \varphi_s$, which is a contradiction since Z'' consists only of supporters of s.

Now we set the canonical interpretation as $Z = X \cup (Y \setminus supp_\Xi(s))$. Observe that there exists $Z' \subseteq att_\Xi(s)$ and $Z'' \subseteq supp_\Xi(s)$ such that $X = Z \setminus Z'$ and $Y = Z \cup Z''$, thus $s \in \mathcal{U}'_\Xi(Z, Z)$ iff $s \in \mathcal{U}'_\Xi(X, Y)$. Since we can construct Z in polynomial time if L^+ and L^- are known and deciding $s \in \mathcal{U}'_\Xi(Z, Z)$ simply amounts to evaluating a formula under a valuation, the first claim follows.

Showing the second claim is similar. Let $Z \subseteq S$, $Z' \subseteq supp_\Xi(s)$ and $Z'' \subseteq att_\Xi(s)$. Then

$$\begin{aligned} & s \in \mathcal{U}''_\Xi(Z, Z) \\ \text{iff } & s \in \mathcal{U}''_\Xi(Z \setminus Z', Z) \\ \text{iff } & s \in \mathcal{U}''_\Xi(Z \setminus Z', Z \cup Z''). \end{aligned}$$

□

Using the generic upper bounds of Theorem 3.18, it is now straightforward to show membership results for BADFs with known link types.

Corollary 5.2. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$, semantics $\sigma \in \{adm, com\}$ and $\tau \in \{cfi, nai\}$. We find that*

- $\mathsf{Ver}^{\mathcal{I}}_\sigma$, $\mathsf{Ver}^{\mathcal{I}}_\tau$ *and* $\mathsf{Ver}^{\mathcal{I}}_{grd}$ *are in* P*;*
- $\mathsf{Ver}^{\mathcal{I}}_{pre}$ *is in* coNP*;*
- $\mathsf{Exists}^{\mathcal{I}}_\sigma$, $\mathsf{Exists}^{\mathcal{I}}_{pre}$ *are in* NP*;*
- $\mathsf{Cred}^{\mathcal{I}}_\tau$ *is in* P*;*
- $\mathsf{Cred}^{\mathcal{I}}_\sigma$ *and* $\mathsf{Cred}^{\mathcal{I}}_{pre}$ *are in* NP*;*
- $\mathsf{Exists}^{\mathcal{I}}_{grd}$, $\mathsf{Cred}^{\mathcal{I}}_{grd}$, $\mathsf{Skept}^{\mathcal{I}}_{grd}$, $\mathsf{Skept}^{\mathcal{I}}_{com}$ *are in* P*;*
- $\mathsf{Skept}^{\mathcal{I}}_{pre}$ *is in* Π^P_2*.*

Proof. *Membership is due to Theorem 3.18 and the complexity bounds of the operators in BADFs in Proposition 5.1, just note that $\Sigma^P_0 = \Pi^P_0 = \mathrm{D}^P_0 = \mathsf{P}$. $\mathsf{Ver}^{\mathcal{I}}_{grd}$ is in $\mathsf{P}^\mathsf{P} = \mathsf{P}$ by Corollary 3.19. For the existence of non-trivial pairs we can simply guess and check in polynomial time for admissible pairs and thus also for complete and preferred semantics.* □

Hardness results straightforwardly carry over from AFs.

Proposition 5.3. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$ and semantics $\sigma \in \{adm, com, pre\}$.*

- $\mathsf{Ver}^{\mathcal{I}}_{pre}$ *is* coNP*-hard;*
- $\mathsf{Exists}^{\mathcal{I}}_\sigma$ *and* $\mathsf{Cred}^{\mathcal{I}}_\sigma$ *are* NP*-hard;*
- $\mathsf{Skept}^{\mathcal{I}}_{pre}$ *is* Π^P_2*-hard.*

Proof. *Hardness results from AFs for these problems carry over to BADFs as for all semantics AFs are a special case of BADFs [11, 40]. The complexities of the problems on AFs for admissible and preferred semantics are shown by Dimopoulos and Torres [22], except for the Π^P_2-completeness result of skeptical preferred semantics, which is shown by Dunne and Bench-Capon [27]. The complete semantics is studied by Coste-Marquis et al. [15].* □

5.1. Conflict-free semantics

For the semantics based on conflict-freeness, it also becomes P-easy to decide whether non-trivial interpretations exist. Recall that by Lemma 3.8, any set of conflict-free interpretations is $\leq_i$-downward-closed. (That is, whenever (X, Y) is conflict-free then any $(X', Y') \leq_i (X, Y)$ is also conflict-free.) This also gives a more intuitive explanation of why $\mathsf{Ver}_{nai}^{\mathcal{I}}$ is in P for $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$: To verify that a conflict-free pair (X, Y) is also naive, we have to verify that the set of pairs

$$\{(X \cup \{s\}, Y), (X, Y \setminus \{s\}) \mid s \in Y \setminus X\}$$

contains no conflict-free pair. This check can be done in polynomial time since there are at most $2 \cdot |S|$ elements in this set and $\mathsf{Ver}_{cfi}^{\mathcal{I}}$ is in P.

Proposition 5.4. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$. $\mathsf{Exists}_{cfi}^{\mathcal{I}}$ and $\mathsf{Exists}_{nai}^{\mathcal{I}}$ are in P.*

Proof. *We first note that the two decision problems coincide by Lemma 3.7. To decide $\mathsf{Exists}_{cfi}^{\mathcal{I}}$ for a given ADF $\Xi = (S, L, C)$, we have to check for each $s \in S$ whether any of the pairs $(\{s\}, S)$ or $(\emptyset, S \setminus \{s\})$ is conflict-free, which can be done in polynomial time by Corollary 5.2. If one of these pairs is conflict-free, the answer is yes; if all pairs where exactly one statement is not undecided are not conflict-free, then there is no non-trivial conflict-free pair. (If there was one, then by Lemma 3.8 there would be a non-trivial conflict-free pair where exactly one statement is true or false.)* □

For skeptical reasoning amongst naive semantics, we can show that the problem remains hard even for bipolar ADFs. This is because we can introduce new statements, which allows us to encode tautology checking of propositional formulas in disjunctive normal form into a bipolar ADF.

Proposition 5.5. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$. $\mathsf{Skept}_{nai}^{\mathcal{I}}$ is coNP-complete.*

Proof. in coNP: *To verify that a statement $s \in S$ does not follow skeptically, we can guess a pair (X, Y) with $s \notin X$ and verify in P that it is naive.*

coNP-hard: *We reduce from tautology checking. Let $\psi = \psi_1 \vee \ldots \vee \psi_n$ be a propositional formula in DNF over vocabulary P. Assume additionally (and without loss of generality) that there is no disjunct ψ_i that contains both p and $\neg p$ for some $p \in P$. (If there is such a disjunct, we can remove it without changing the models of ψ.) We construct a bipolar ADF $D = (S, L, C)$ as follows:*

$$\begin{aligned} S &= P \cup \{z, d_1, \ldots, d_n\} \\ \varphi_p &= p && (p \in P) \\ \varphi_{d_i} &= \psi_i && (1 \leq i \leq n) \\ \varphi_z &= d_1 \vee \ldots \vee d_n \end{aligned}$$

We show that z is contained in all naive pairs of D iff ψ is a tautology.

"if": *Let ψ be a tautology. Given an $M \subseteq P$, define a set N_M as follows:*

$$N_M = M \cup \{d_i \mid M \models \psi_i\} \cup \{z\}$$

We show that for each $M \subseteq P$, the pair $\bar{m}_M = (N_M, N_M)$ is naive, and these are the only naive pairs. We first observe that each such pair is two-valued, and thus the two *operators (approximate and ultimate) coincide on it, furthermore we need only show conflict-freeness to show naivety. It is clear that $\bar{m}_M$ is conflict-free with respect to all $p \in P$. For $1 \leq i \leq n$, conflict-freeness of $\bar{m}_M$ with respect to d_i follows by definition. Since ψ is a tautology, there is at least one d_i in each N_M, and $z \in N_M$ is justified. Assume there were another naive pair (X, Y) with $z \notin X$. First of all, each naive pair must constitute a two-valued interpretation of the statements in P, for otherwise the $\leq_i$-maximality condition would be violated. Now this enforces a fixed truth value for $d_1, \ldots, d_n$ and thus also for z. As argued above, $z \in N_M$ necessarily holds.*

"only if": *Let ψ be refutable. Then there is an $M \subseteq P$ such that we find $M \not\models \psi_i$ for all $1 \leq i \leq n$. We show that the pair $\bar{m} = (M, M)$ is naive for approximate and ultimate operator. Clearly by presumption, for all $1 \leq i \leq n$ we find that $\psi_i^{(M,M)}$ is a Boolean expression that evaluates to false, so having $d_i \notin M$ in the upper bound of the pair $\bar{m}$ is justified. Finally, $\varphi_z = d_1 \vee \ldots \vee d_n$ also evaluates to false, thus justifying $z \notin M$. Thus there is a naive pair $(X, Y) = (M, M)$ with $z \notin X$.* □

Notably, this result is the only case in which bipolar ADFs are (potentially) more complex than AFs, as in the latter skeptical reasoning over naive pairs can be done in polynomial time [15].[9]

5.2. Two-valued semantics

Regarding BADFs and two-valued semantics we first show that there is no hope that the existence problems for approximate and ultimate two-valued stable models coincide as there are cases when the semantics differ.

Example 5.1. Consider the BADF $F = (S, L, C)$ with statements $S = \{a, b, c\}$ and acceptance formulas $\varphi_a = \mathbf{t}$, $\varphi_b = a \vee c$ and $\varphi_c = a \vee b$. The only two-valued supported model is (S, S) where all statements are true. This pair is also an ultimate two-valued stable model, since $\mathcal{U}'_F(\emptyset, S) = \{a\}$, and both $\varphi_b^{(\{a\},S)} = \mathbf{t} \vee c$ and $\varphi_c^{(\{a\},S)} = \mathbf{t} \vee b$ are tautologies, whence we have $\mathcal{U}'_F(\{a\}, S) = S$. However, (S, S) is not an approximate two-valued stable model: although $\mathcal{G}'_F(\emptyset, S) = \{a\}$, then $\mathcal{G}'_F(\{a\}, S) = \{a\}$ since the partially evaluated formulas $\varphi_b^{(\{a\},S)}$ and $\varphi_c^{(\{a\},S)}$ contain free variables. We thus cannot reconstruct the upper bound S and F has no approximate two-valued stable models.

So approximate and ultimate two-valued stable model semantics are indeed different. However, we can show that the respective existence problems have the same complexity.

Proposition 5.6. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$ and semantics $\sigma \in \{2su, 2st\}$. $\mathsf{Ver}_\sigma^\mathcal{I}$ is in P; $\mathsf{Exists}_\sigma^\mathcal{I}$ is NP-complete.*

Proof. *Membership carries over – for supported models from [11, Proposition 5], for approximate stable models from Theorem 4.24. For membership for ultimate stable models, we can use Proposition 5.1 to adapt the decision procedure of Proposition 4.23. In any case, hardness carries over from AFs [22].* □

For credulous and skeptical reasoning over the two-valued semantics, membership is straightforward and hardness again carries over from argumentation frameworks.

[9] To check whether an argument a is skeptically accepted for naive semantics, we only have to check whether all its attackers are self-attacking: if there is a b that attacks a and is not self-attacking, then the set $\{b\}$ is conflict-free, thus there exists a naive set $N \supseteq \{b\}$ with $a \notin N$.

Corollary 5.7. *Let $\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$ and semantics $\sigma \in \{\mathit{2su}, \mathit{2st}\}$.* $\mathsf{Cred}_\sigma^\mathcal{I}$ *is* NP*-complete;* $\mathsf{Skept}_\sigma^\mathcal{I}$ *is* coNP*-complete.*

$\mathcal{I} \in \{\mathcal{BG}, \mathcal{BU}\}$, σ	conflict-free	naive	admissible	complete	preferred	grounded	model	stable model
$\mathsf{Ver}_\sigma^\mathcal{I}$	in P (Corollary 5.2)	in P (Corollary 5.2)	in P (Corollary 5.2)	in P (Corollary 5.2)	coNP-c (Corollary 5.2, Proposition 5.3)	in P (Corollary 5.2)	in P (Proposition 5.6)	in P (Proposition 5.6)
$\mathsf{Exists}_\sigma^\mathcal{I}$	in P (Proposition 5.4)	in P (Proposition 5.4)	NP-c (Corollary 5.2, Proposition 5.3)	NP-c (Corollary 5.2, Proposition 5.3)	NP-c (Corollary 5.2, Proposition 5.3)	in P (Corollary 5.2)	NP-c (Proposition 5.6)	NP-c (Proposition 5.6)
$\mathsf{Cred}_\sigma^\mathcal{I}$	in P (Corollary 5.2)	in P (Corollary 5.2)	NP-c (Corollary 5.2, Proposition 5.3)	NP-c (Corollary 5.2, Proposition 5.3)	NP-c (Corollary 5.2, Proposition 5.3)	in P (Corollary 5.2)	NP-c (Corollary 5.7)	NP-c (Corollary 5.7)
$\mathsf{Skept}_\sigma^\mathcal{I}$	trivial	coNP-c (Proposition 5.5)	trivial	in P (Corollary 5.2)	Π_2^P-c (Corollary 5.2, Proposition 5.3)	in P (Corollary 5.2)	coNP-c (Corollary 5.7)	coNP-c (Corollary 5.7)

Table 3: Complexity results for semantics of bipolar Abstract Dialectical Frameworks.

6. Discussion

In this paper we studied the computational complexity of abstract dialectical frameworks using approximation fixpoint theory. We showed numerous novel results for two families of ADF semantics, the approximate and ultimate semantics, which are themselves inspired by argumentation and AFT. We showed that in most cases the complexity increases by one level of the polynomial hierarchy compared to the corresponding reasoning tasks on AFs. Notable differences between the two families of semantics lie in the stable model semantics and in semantics based on conflict-freeness, where the approximate version is easier than its ultimate counterpart. For the restricted, yet powerful class of bipolar ADFs we proved that for the corresponding reasoning tasks AFs and BADFs have (almost) the same complexity, with the single exception of skeptical reasoning among naive pairs. This suggests that many types of relations between arguments can be introduced without increasing the worst-time complexity. On the other hand, our results again emphasize that arbitrary (non-bipolar) ADFs cannot be compiled into equivalent Dung AFs in deterministic polynomial time, unless the polynomial hierarchy collapses to the first level. Under the same assumption, ADFs cannot be implemented directly with methods that are typically applied to AFs, for example answer-set programming [31].

Our results on the complexity of bipolar ADFs led to our extending the ADF system DIAMOND [32] with specialized implementation techniques for bipolar ADFs. In the future, we also plan to accommodate the approximate semantics family into DIAMOND. In another direction of work, QBF encodings for general ADFs were developed and implemented in the system QADF [21]. For further future work several promising directions are possible. Studying easier fragments of ADFs as well as parameterized complexity analysis can lead to efficient decision procedures, as is witnessed for AFs [30, 29]. We also deem it auspicious to use alternative representations of acceptance conditions, for instance by employing techniques from knowledge compilation [16].

In recent related work, Gaggl et al. [34] analysed the computational complexity of naive-based ADF semantics as defined by Gaggl and Strass [33]. A detailed comparison of the two types of semantics and their respective complexities is left for future work. A complexity analysis of other useful AF semantics would also reveal further insights, for example semi-stable semantics [14] or ideal semantics [25, 26]. Furthermore in [39, 38] several extension-based semantics for ADFs are proposed and a complexity analysis would be interesting. Bogaerts et al. [6] recently defined a new semantics for ADFs (the grounded fixpoint semantics) that is not unlike (ultimate) stable models. This similarity is also backed up by their initial complexity analysis, and a more detailed study might be interesting future work. Another recent new ADF semantics are the F-stable models defined by Alviano and Faber [1]; the complexity of that semantics is as yet unexplored.

For semantical analysis, it would be useful to consider principle-based evaluations for ADFs [2]. Furthermore it appears natural to compare (ultimate) ADF semantics and ultimate logic programming semantics [20] in approximation fixpoint theory, in particular with respect to computational complexity. Finally, we want to apply the general operator splitting results of Vennekens et al. [46] to abstract argumentation and compare them to the stand-alone results obtained for AFs [3] and ADFs [36].

Acknowledgements. This research was supported by DFG (project BR 1817/7-1) and FWF (project I1102). We thank several anonymous reviewers for helpful feedback.

References

[1] Mario Alviano and Wolfgang Faber. Stable model semantics of abstract dialectical frameworks revisited: A logic programming perspective. In *Proceedings of the Twenty-Fourth*

International Joint Conference on Artificial Intelligence (IJCAI), Buenos Aires, Argentina, July 2015. In press.

[2] Pietro Baroni and Massimiliano Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.

[3] Ringo Baumann. Splitting an argumentation framework. In James P. Delgrande and Wolfgang Faber, editors, *Proceedings of the Eleventh International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, volume 6645 of *LNCS*, pages 40–53. Springer, 2011.

[4] Ringo Baumann, Wolfgang Dvořák, Thomas Linsbichler, Hannes Strass, and Stefan Woltran. Compact Argumentation Frameworks. In Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, editors, *Proceedings of the Twenty-First European Conference on Artificial Intelligence (ECAI)*, pages 69–74. IOS Press, 2014.

[5] Trevor J. M. Bench-Capon and Paul E. Dunne. Argumentation in Artificial Intelligence. *Artificial Intelligence*, 171(10-15):619–641, July 2007. ISSN 0004-3702.

[6] Bart Bogaerts, Joost Vennekens, and Marc Denecker. Grounded fixpoints and their applications in knowledge representation. *Artificial Intelligence*, 224:51–71, 2015.

[7] Nicolas Bourbaki. Sur le théorème de Zorn. *Archiv der Mathematik*, pages 434–437, 1949/50.

[8] Gerhard Brewka and Thomas F. Gordon. Carneades and Abstract Dialectical Frameworks: A Reconstruction. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Proceedings of the Third International Conference on Computational Models of Argument (COMMA 2010)*, volume 216 of *FAIA*, pages 3–12. IOS Press, 2010.

[9] Gerhard Brewka and Stefan Woltran. Abstract Dialectical Frameworks. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczyński, editors, *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning (KR 2010)*, pages 102–111. AAAI Press, 2010.

[10] Gerhard Brewka, Paul E. Dunne, and Stefan Woltran. Relating the Semantics of Abstract Dialectical Frameworks and Standard AFs. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pages 780–785. IJCAI/AAAI, 2011.

[11] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract Dialectical Frameworks Revisited. In Francesca Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI 2013)*, pages 803–809. IJCAI/AAAI, 2013.

[12] Gerhard Brewka, Sylwia Polberg, and Stefan Woltran. Generalizations of Dung frameworks and their role in formal argumentation. *IEEE Intelligent Systems*, 29(1), 2013. Special Issue on Representation and Reasoning.

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

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

[15] Sylvie Coste-Marquis, Caroline Devred, and Pierre Marquis. Symmetric Argumentation Frameworks. In Lluis Godo, editor, *Proceedings of the Eighth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, volume 3571 of *LNCS*, pages 317–328. Springer, 2005.

[16] Adnan Darwiche and Pierre Marquis. A Knowledge Compilation Map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.

[17] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, second edition, 2002.

[18] Marc Denecker, Victor W. 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.

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

[20] Marc Denecker, Victor W. Marek, and Miroslaw Truszczyński. Ultimate approximation and its application in nonmonotonic knowledge representation systems. *Information and Computation*, 192(1):84–121, 2004.

[21] Martin Diller, Johannes P. Wallner, and Stefan Woltran. Reasoning in Abstract Dialectical Frameworks using quantified Boolean formulas. In Simon Parsons, Nir Oren, and Chris Reed, editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, volume 266 of *FAIA*, pages 241–252. IOS Press, 2014.

[22] Yannis Dimopoulos and Alberto Torres. Graph Theoretical Structures in Logic Programs and Default Theories. *Theoretical Computer Science*, 170(1-2):209–244, 1996.

[23] Yannis Dimopoulos, Bernhard Nebel, and Francesca Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141(1/2): 57–78, 2002.

[24] Phan Minh Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. *Artificial Intelligence*, 77 (2):321–358, 1995.

[25] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10):642–674, 2007.

[26] Paul E. Dunne. The computational complexity of ideal semantics. *Artificial Intelligence*, 173(18):1559–1591, 2009.

[27] Paul E. Dunne and Trevor J. M. Bench-Capon. Coherence in Finite Argument Systems. *Artificial Intelligence*, 141(1/2):187–203, 2002.

[28] Paul E. Dunne and Michael Wooldridge. Complexity of abstract argumentation. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 85–104. Springer, 2009.

[29] Wolfgang Dvořák, Sebastian Ordyniak, and Stefan Szeider. Augmenting tractable fragments of abstract argumentation. *Artificial Intelligence*, 186:157–173, 2012.

[30] Wolfgang Dvořák, Matti Järvisalo, Johannes P. Wallner, and Stefan Woltran. Complexity-Sensitive Decision Procedures for Abstract Argumentation. *Artificial Intelligence*, 206:53–78, 2014.

[31] Uwe Egly, Sarah A. Gaggl, and Stefan Woltran. Answer-set programming encodings for argumentation frameworks. *Argument and Computation*, 1(2):147–177, 2010.

[32] Stefan Ellmauthaler and Hannes Strass. The DIAMOND System for Computing with Abstract Dialectical Frameworks. In Simon Parsons, Nir Oren, and Chris Reed, editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, volume 266 of *FAIA*, pages 233–240. IOS Press, 2014.

[33] Sarah A. Gaggl and Hannes Strass. Decomposing Abstract Dialectical Frameworks. In Simon Parsons, Nir Oren, and Chris Reed, editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, volume 266 of *FAIA*, pages 281–292. IOS Press, 2014.

[34] Sarah A. Gaggl, Sebastian Rudolph, and Hannes Strass. On the computational complexity of naive-based semantics for abstract dialectical frameworks. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 2015. In press.

[35] 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.

[36] Thomas Linsbichler. Splitting abstract dialectical frameworks. In Simon Parsons, Nir Oren, and Chris Reed, editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, volume 266 of *FAIA*, pages 357–368. IOS Press, 2014.

[37] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994. ISBN 978-0-201-53082-7.

[38] Sylwia Polberg. Extension-based Semantics of Abstract Dialectical Frameworks. In Sébastien Konieczny and Hans Tompits, editors, *Proceedings of the Fifteenth International Workshop on Non-Monotonic Reasoning (NMR 2014)*, pages 273–282, 2014.

[39] Sylwia Polberg, Johannes P. Wallner, and Stefan Woltran. Admissibility in the Abstract Dialectical Framework. In João Leite, Tran Cao Son, Paolo Torroni, Leon van der Torre, and Stefan Woltran, editors, *Proceedings of the 14th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA 2013)*, volume 8143 of *LNAI*, pages 102–118. Springer, 2013.

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

[41] Hannes Strass. Instantiating knowledge bases in abstract dialectical frameworks. In João Leite, Tran Cao Son, Paolo Torroni, Leon van der Torre, and Stefan Woltran, editors, *Proceedings of the Fourteenth International Workshop on Computational Logic in Multi-Agent Systems (CLIMA 2013)*, volume 8143 of *LNAI*, pages 86–101. Springer, 2013.

[42] Hannes Strass. The relative expressiveness of abstract argumentation and logic programming. In Sven Koenig and Blai Bonet, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI 2015)*, pages 1625–1631, 2015.

[43] Hannes Strass. Instantiating rule-based defeasible theories in abstract dialectical frameworks and beyond. *Journal of Logic and Computation*, 2015. doi: 10.1093/logcom/exv004.

[44] Hannes Strass and Johannes P. Wallner. Analyzing the Computational Complexity of Abstract Dialectical Frameworks via Approximation Fixpoint Theory. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proceedings of the Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning (KR 2014)*, pages 101–110. AAAI Press, 2014.

[45] Bas Van Gijzel and Henry Prakken. Relating Carneades with abstract argumentation. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pages 1113–1119. IJCAI/AAAI, 2011.

[46] Joost Vennekens, David Gilis, and Marc Denecker. Splitting an operator: Algebraic modularity results for logics with fixpoint semantics. *ACM Transactions on Computational Logic*, 7(4):765–797, 2006.

[47] Adam Wyner, Trevor J. M. Bench-Capon, and Paul E. Dunne. On the instantiation of knowledge bases in abstract argumentation frameworks. In João Leite, Tran Cao Son, Paolo Torroni, Leon van der Torre, and Stefan Woltran, editors, *Proceedings of the Fourteenth International Workshop on Computational Logic in Multi-Agent Systems (CLIMA 2013)*, volume 8143 of *LNAI*, pages 34–50. Springer, 2013.