Acceptance in Incomplete Argumentation Frameworks

Dorothea Baumeister[a], Matti Järvisalo[b], Daniel Neugebauer[a], Andreas Niskanen[b], Jörg Rothe[a]

[a] *Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, Germany*
[b] *Helsinki Institute for Information Technology HIIT, Department of Computer Science, University of Helsinki, Finland*

Abstract

Abstract argumentation frameworks (AFs), originally proposed by Dung, constitute a central formal model for the study of computational aspects of argumentation in AI. Credulous and skeptical acceptance of arguments in a given AF are well-studied problems both in terms of theoretical analysis—especially computational complexity—and the development of practical decision procedures for the problems. However, AFs make the assumption that all attacks between arguments are certain (i.e., present attacks are known to exist, and missing attacks are known to not exist), which can in various settings be a restrictive assumption. A generalization of AFs to incomplete AFs was recently proposed as a formalism that allows the representation of both uncertain attacks and uncertain arguments in AFs. In this article, we explore the impact of allowing for modeling such uncertainties in AFs on the computational complexity of natural generalizations of acceptance problems to incomplete AFs under various central AF semantics. Complementing the complexity-theoretic analysis, we also develop the first practical decision procedures for all of the NP-hard variants of acceptance in incomplete AFs. In terms of complexity analysis, we establish a full complexity landscape, showing that depending on the variant of acceptance and property/semantics, the complexity of acceptance in incomplete AFs ranges from polynomial-time decidable to completeness for Σ_3^p. In terms of algorithms, we show through an extensive empirical evaluation that an implementation of the proposed decision procedures, based on boolean satisfiability (SAT) solving, is effective in deciding variants of acceptance under uncertainties. We also establish conditions for what type of atomic changes are guaranteed to be redundant from the perspective of preserving extensions of completions of incomplete AFs, and show that the results allow for considerably improving the empirical efficiency of the proposed SAT-based counterexample-guided abstraction refinement algorithms for acceptance in incomplete AFs for problem variants with complexity beyond NP.

Keywords: Abstract argumentation, incomplete knowledge, incomplete argumentation frameworks, computational complexity, decision procedures, empirical evaluation

1. Introduction

The study of computational aspects of argumentation is a topical area of artificial intelligence research. With strong connections to other forms of nonmonotonic reasoning, abstract argumentation frameworks [41] (AFs) provide a central formal model for the study of argumentation in AI. Argumentation frameworks take the form of directed graphs, where the nodes represent abstract arguments, and directed edges form an attack relation between arguments.

While originally AFs make the assumption that all attacks between arguments are *certain*, in various settings such an assumption turns out to be restrictive. In an answer to bypass this restriction, a generalization of AFs to *incomplete argumentation frameworks* (IAFs) was recently proposed [16, 17], bringing together earlier-proposed ideas of a generalization of AFs to partial argumentation frameworks [33, 14] (integrating uncertainty about the existence of attacks into AFs) and the integration of uncertainty about the existence of arguments into AFs [18]. This article contributes to the study of incomplete argumentation frameworks from different computational perspectives. In particular, we establish the computational complexity of central reasoning problems and their variants in the context of IAFs under various argumentation semantics, and develop practical decision procedures for NP-hard reasoning tasks in IAFs.

A key motivation behind incomplete argumentation frameworks is that they allow for representing unquantified structural uncertainty, i.e., uncertainty about the existence of particular attacks or arguments without any specified probability of existence. Uncertainty is inherently present in argumentative scenarios. In particular, real arguments as presented by people in dialogues are usually enthymemes [56, 3], which represent explicitly only a part of the underlying knowledge (i.e., the premises and the claim of the argument). In discussions one can typically assume a common knowledge base which one can use to construct arguments in a partially implicit manner. However, assumptions on the common knowledge base can be different for different agents, which means that the AFs constructed by the agents may have different arguments and attacks [63]. The ASPIC$^+$ framework [66] and the encoding proposed by Wyner et al. [84] are two examples of specific instantiation methods that could translate uncertainty in an underlying knowledge base to structural uncertainty represented by an incomplete argumentation framework (see Section 8 on related work for more details).

Additionally, incomplete argumentation frameworks allow for modeling different types of real-world application scenarios. As an example, consider several agents, each with their own AF representing their subjective view, and the problem of merging these AFs [33] (again, see Section 8 for more examples on interesting application scenarios). While the AFs may coincide in terms of the presence of some arguments and attacks (which can be considered definite), it is likely that agents partially disagree on the existence of specific arguments or attacks, or that specific arguments or attacks are simply not represented in the AFs of some agents. Such arguments and attacks may be considered uncertain. By merging the agents' AFs into a single incomplete argumentation framework by taking the union of all arguments and attacks and specifying elements as definite if they occur in each AF, and otherwise as uncertain, the resulting single incomplete AF allows for reasoning about, e.g., whether some or all agents find an argument acceptable. For more details about various approaches to collective acceptability and specific methods for structural aggregation of AFs, we refer to the forthcoming handbook chapter by Baumeister et al. [13].

An incomplete argumentation framework can be seen as a representation of a set of possible worlds, called *completions*, each of which is a standard argumentation framework that shares all definite elements of the incomplete framework and where each of its uncertain elements is either included or excluded. Existing criteria for argumentation frameworks can then be generalized to incomplete argumentation frameworks by either asking whether they are satisfied *possibly* (in at least one completion) or *necessarily* (in all completions), i.e., whether the uncertainty either can or must be resolved in a way that satisfies the conditions of the given criterion. The answer may help with decisions in strategic scenarios, where the uncertainty represents possible moves. In scenarios where uncertainty represents missing information, the preliminary answer may be sufficient for the task at hand, removing the need to actually resolve the uncertainty.

In this article, we focus on the central reasoning problems of *credulous* and *skeptical acceptance* for incomplete argumentation frameworks. Credulous and skeptical acceptance are today well-understood when it comes to (standard) argumentation frameworks (i.e., AFs without any uncertainties on the existence of arguments and attacks). In terms of standard AFs, acceptance consists of asking—parameterized by a semantics and for a given argumentation framework and an argument in that framework—either whether that argument is in *at least one* extension (for credulous acceptance) or in *all* extensions (for skeptical acceptance) of the framework with respect to the semantics. Generalizing acceptance in a natural way to IAFs, we focus on variants of the following four problem combinations.

- **Possible Credulous Acceptance (PCA)**:
  Is there *any* way to accept the given argument?
- **Necessary Credulous Acceptance (NCA)**:
  Is the given argument in at least one extension, *regardless* of how the uncertainty is resolved?
- **Possible Skeptical Acceptance (PSA)**:
  Can the uncertainty be resolved in such a way that the given argument is in *all* extensions?
- **Necessary Skeptical Acceptance (NSA)**:
  Is the given argument absolutely *guaranteed* to be accepted?

A "no" answer to PCA indicates that the target argument is a hopeless case and will never be accepted, while a "yes" answer to NSA guarantees that it will be accepted under all circumstances. If the goal is to have the target argument credulously accepted, then a "yes" answer to NCA ensures that this goal is satisfied, whatever completion or extension is chosen. If the goal is to have the target skeptically accepted, then the answer to PSA indicates whether that goal can be achieved by choosing the right completion. All of these answers provide information that is valuable even when the uncertainty cannot be resolved.

Note that in these forms, however, the acceptance problems may show undesired behavior if sets of acceptable arguments (extensions) are empty, or if no extension exists for a particular semantics: The skeptical acceptance (SA) problem is trivial for all semantics that always accept the empty set, since then no argument can ever be in *all* extensions. Further, all semantics that do not guarantee the existence of an extension allow for cases where no argument is credulously accepted, but simultaneously, all arguments are skeptically accepted, which may be counterintuitive. These issues suggest further variants of acceptance for AFs, namely, restricting the semantics to nonempty extensions and further requiring the existence of at least one extension in order to give a "yes" answer for this variant of skeptical acceptance. While SA alone indicates whether the target argument is among the "best-accepted" arguments in the AF with respect to a given semantics, the refined variant ExSA—formally defined as SA with the additional condition that an extension exists—indicates whether the target argument is actually and ultimately accepted in the AF with respect to the semantics. We incorporate these refinements by defining the nonempty restrictions $\mathrm{CF}_{\neq\emptyset}$ and $\mathrm{AD}_{\neq\emptyset}$ of conflict-freeness (CF) and admissibility (AD) (to be formally defined in Section 2) and by generalizing the ExSA problem to PExSA and NExSA for IAFs.

In terms of complexity analysis, we establish a full complexity landscape of all of the mentioned variants of acceptance problems in incomplete argumentation frameworks, considering a range of central argumentation properties and semantics: standard and nonempty conflict-freeness and admissibility; and stable, complete, grounded, and preferred semantics. Naturally, as IAFs generalize standard AFs, the complexity of acceptance problems in IAFs is always at least as high as that of corresponding problems in AFs. It turns out that, depending on the acceptance problem and the property or semantics used, the complexity of acceptance in IAFs ranges from polynomial-time decidable to completeness for Σ_3^p, a complexity class in the third level of the polynomial hierarchy [65, 80]. In contrast, the computational complexity of acceptance problems in standard AFs [38, 43] under the same semantics ranges from polynomial-time decidability to completeness for Π_2^p, i.e., for certain problem variants we have a one-level jump in complexity in terms of the polynomial hierarchy when moving from AFs to IAFs. Intuitively, this complexity jump arises from alternating quantifiers in the respective problem definitions.

While we establish polynomial-time decidability—directly implying practical specialized algorithms—for specific problem variants and semantics, most of the variants of acceptance in incomplete argumentations turn out to be hard for NP, coNP, or even a class higher in the polynomial hierarchy. Motivated by the success of practical boolean satisfiability (SAT) [21] based decision procedures developed for acceptance in standard AFs [46, 29], we present the first SAT-based approach to reasoning about acceptance in incomplete AFs. Complementing the complexity results, our SAT-based algorithms cover all of the considered variants of acceptance in incomplete AFs and semantics considered in our complexity analysis. In particular, for the problem variants that turn out to be complete for the first level of the polynomial hierarchy, generalizing SAT encodings of AF semantics [20] to cover acceptance in incomplete AFs, we present direct SAT encodings which allow for deciding acceptance with a single call to a SAT solver. For those problem variants that turn out to be complete for the second or third level of the polynomial hierarchy, we develop SAT-based counterexample-guided abstraction refinement procedures, making incremental use of a SAT solver to decide acceptance in an iterative manner. We also present results from an extensive empirical study of our implementation of all of the SAT-based algorithms presented, showing the effectiveness of the approach.

Bridging theory and practice, extending on earlier results on the persistence of extensions under atomic changes to AFs [77], we also establish conditions for what type of atomic changes are guaranteed to be redundant from the perspective of preserving extensions of completions of IAFs. While of interest on their own, this analysis proves central as a basis of SAT-based counterexample-guided abstraction refinement (CEGAR) algorithms for IAFs. In particular, we show empirically that the more in-depth analysis of atomic changes to the uncertain part under which an extension persists gives noticeably stronger refinements for the SAT-based CEGAR algorithms for deciding acceptance in incomplete AFs, evidenced in practice by noticeably improved runtimes.

The rest of this article is organized as follows. In Section 2, we give the required formal background regarding abstract argumentation frameworks, incomplete argumentation frameworks, computational complexity theory, and SAT solvers. We give a full study of the computational complexity of acceptance problems for incomplete argumentation frameworks in Section 3. In Section 4, we provide results on which atomic changes to the uncertain part of an incomplete argumentation framework are redundant concerning the acceptability of a given set of arguments. For all possible and necessary acceptance problem variants that are complete for complexity classes in the first, second, or third level of the polynomial hierarchy, we propose SAT encodings in Section 5 and SAT-based algorithms in Section 6. Section 7 reports on an extensive empirical evaluation of our implementation of the algorithms. Section 8 contains a detailed comparison of the model of incomplete argumentation framework to other models that represent uncertainty in abstract argumentation, and Section 9 summarizes our contribution and suggests future work directions.

Some of the results presented in this article have been preliminarily presented at AAAI 2020 [71] and COMMA 2018 [15]. This article considerably expands and extends these preliminary conference versions by including all formal proofs in full (omitted from the preliminary versions); by presenting further non-trivial SAT-based algorithms—in particular, SAT encodings for complete and grounded semantics, and detailed SAT-based algorithms for necessary credulous acceptance under admissible and stable semantics, and for both necessary and possible skeptical acceptance under preferred semantics; by reporting results from a considerably extended empirical evaluation; and by additional discussion and illustrative examples for self-containment.

2. Preliminaries

In this section, we first provide the needed notions from standard abstract argumentation (where we have no uncertainty about the existence of arguments or attacks), define the semantics we will consider, and formally define the notions and associated decision problems of credulous and skeptical acceptance for them as well as a certain natural restriction of skeptical acceptance that is due to Dunne and Wooldridge [44]. Then we will formally describe the incomplete argumentation frameworks due to Baumeister et al. [16, 17] and will define variants of credulous and skeptical acceptance (and of the above-mentioned restriction of the latter as well) that model whether a given argument is accepted possibly or necessarily for a given semantics.

2.1. Argumentation Frameworks

An *argumentation framework* $AF = \langle \mathscr{A}, \mathscr{R} \rangle$ consists of a finite set $\mathscr{A}$ of *arguments* and a binary *attack relation* $\mathscr{R} \subseteq \mathscr{A} \times \mathscr{A}$ on the arguments, where $(a,b) \in \mathscr{R}$ indicates that *a attacks b*.

Example 1. An AF $\langle \mathscr{A}, \mathscr{R} \rangle$ can be represented as a directed graph by identifying $\mathscr{A}$ with the set of nodes and $\mathscr{R}$ with the set of directed edges of this graph. Figure 1 displays an argumentation framework with arguments $\mathscr{A} = \{a,b,c,d,e,f,g\}$ and attacks $\mathscr{R} = \{(a,b),(b,c),(c,b),(c,e),(d,c),(d,f),(e,d),(f,d),(g,e),(g,g)\}$.

A set $A \subseteq \mathscr{A}$ is *conflict-free* (CF) if $(a,b) \notin \mathscr{R}$ for all $a,b \in A$. An argument $a \in \mathscr{A}$ is *defended by a set $A \subseteq \mathscr{A}$ of arguments in AF* if, for each attacker $b \in \mathscr{A}$ of a with $(b,a) \in \mathscr{R}$, there is a defender $d \in A$ of a with $(d,b) \in \mathscr{R}$. The *characteristic function of AF*, $F_{AF} : 2^{\mathscr{A}} \to 2^{\mathscr{A}}$, outputs all arguments defended by a given set, i.e., $F_{AF}(A) = \{a \in \mathscr{A} \mid a \text{ is defended by } A \text{ in } AF\}$. F_{AF}^{k} denotes the k-fold composition of F_{AF}, and F_{AF}^{*} denotes its infinite composition. A conflict-free set $A \subseteq \mathscr{A}$ is *admissible* (AD) if $A \subseteq F_{AF}(A)$, i.e., if every argument in A is defended by A in AF. These notions allow the definition of the following *semantics* for argumentation frameworks (see Baroni et al. [5] for a detailed introduction) that each provide an individual criterion to determine acceptable sets of arguments.

Definition 2. Let $AF = \langle \mathscr{A}, \mathscr{R} \rangle$ be an argumentation framework and let $A \subseteq \mathscr{A}$ be a conflict-free set of arguments.

- A is *complete* (CP) if it is a fixed point of the characteristic function of AF, i.e., if $A = F_{AF}(A)$.
- A is *grounded* (GR) if it is the unique least fixed point of the characteristic function of AF, i.e., if $A = F_{AF}^{*}(\emptyset)$.

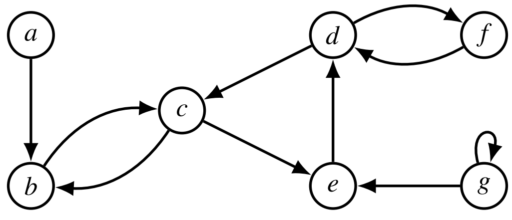
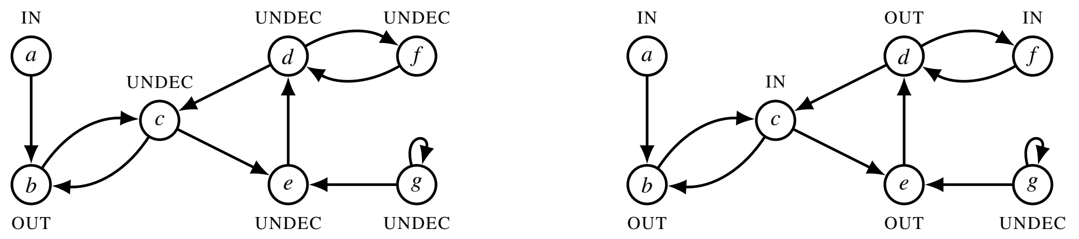


Figure 1: Graph representation of the argumentation framework in Example 1.



(a) Grounded extension. (b) Preferred extension.

Figure 2: Visualization of both complete extensions in the AF from Example 1, where arguments are labeled IN if they are in the extension, labeled OUT if they are attacked by IN arguments, and labeled UNDEC otherwise.

- A is *preferred* (PR) if A is a set-maximal admissible set.
- A is *stable* (ST) if for every $b \in \mathscr{A} \setminus A$, there is an $a \in A$ with $(a,b) \in \mathscr{R}$.

A set of arguments that satisfies the conditions of a semantics is called an *extension* of the argumentation framework with respect to that semantics. Every stable extension is preferred, every preferred extension is complete, every complete extension is admissible, and every admissible set is conflict-free. Further, the unique grounded extension is complete, and coincides with the intersection of all complete extensions. There are argumentation frameworks that have no stable extension, while the other semantics defined above guarantee the existence of at least one extension.

Example 3. We determine all extensions of the argumentation framework in Figure 1 from Example 1. Its conflict-free sets of arguments are $\{a,c,f\}$, $\{a,d\}$, $\{a,f,e\}$, $\{b,d\}$, $\{b,f,e\}$, and all their subsets. Of these, only $\emptyset$, $\{a\}$, $\{f\}$, $\{a,f\}$, $\{c,f\}$, and $\{a,c,f\}$ defend all their members (and are therefore admissible), since argument a is unattacked, argument f defends itself against its only attacker d, and argument c defends itself against b and is defended by f against d. To determine the complete extensions, we can check which of the admissible sets are fixed points of F_{AF}. We have $F_{AF}(\emptyset) = \{a\}$, $F_{AF}(\{a\}) = \{a\}$, $F_{AF}(\{f\}) = \{a,f\}$, $F_{AF}(\{a,f\}) = \{a,c,f\}$, $F_{AF}(\{c,f\}) = \{a,c,f\}$, and $F_{AF}(\{a,c,f\}) = \{a,c,f\}$. Thus $\{a\}$ and $\{a,c,f\}$ are the only complete extensions of AF. Since $F_{AF}(\emptyset) = \{a\}$ and $F_{AF}(\{a\}) = \{a\}$, we also know that $\{a\}$ is the grounded extension of AF. The only set-maximal admissible set is $\{a,c,f\}$, which thus is the only preferred extension of AF. AF has no stable extension, because no conflict-free set has a chance to attack the self-attacking argument g.

Both complete extensions of this example are displayed in Figure 2 using the labeling representation by Caminada [26], where arguments are labeled IN if they are in the extension, labeled OUT if they are attacked by IN arguments, and labeled UNDEC otherwise.

We investigate problems concerning the acceptability of individual arguments in an argumentation framework. The most established notion of acceptability for single arguments proposed by Dunne and Bench-Capon [43] is derived from their membership in extensions. For a given $\mathbf{s} \in \{\text{CF}, \text{AD}, \text{CP}, \text{GR}, \text{PR}, \text{ST}\}$, an argument a can only be considered acceptable if a is contained in at least one $\mathbf{s}$ extension (called *credulous* acceptance), and a is ultimately accepted if it is contained in all $\mathbf{s}$ extensions (called *skeptical* acceptance). The following decision problems formalize these notions.

s-CREDULOUS-ACCEPTANCE (**s**-CA)	
Given:	An argumentation framework $\langle \mathscr{A}, \mathscr{R} \rangle$ and an argument $a \in \mathscr{A}$.
Question:	Is there an **s** extension $\mathscr{E}$ of $\langle \mathscr{A}, \mathscr{R} \rangle$ with $a \in \mathscr{E}$?

s-SKEPTICAL-ACCEPTANCE (**s**-SA)	
Given:	An argumentation framework $\langle \mathscr{A}, \mathscr{R} \rangle$ and an argument $a \in \mathscr{A}$.
Question:	For all **s** extensions $\mathscr{E}$ of $\langle \mathscr{A}, \mathscr{R} \rangle$, does $a \in \mathscr{E}$ hold?

The question of the **s**-Skeptical-Acceptance problem is equivalent to asking whether each set of arguments that does *not* include a is *not* **s** in $\langle \mathscr{A}, \mathscr{R} \rangle$. This alternative formulation moves the verification part of the problem (which checks whether a set of arguments is an extension) from the scope of the quantifier to the predicate after the quantifier, which will allow us to directly derive upper complexity bounds later.

The Skeptical-Acceptance problem exhibits two types of special behavior that we need to address. Firstly, the answer to **s**-Skeptical-Acceptance is trivially "no" if the empty set always satisfies **s**—this is the case for $\mathbf{s} \in \{\mathrm{CF}, \mathrm{AD}\}$ among the properties/semantics that we use in this article. More meaningful results can be obtained by excluding the empty set for acceptance problems. Therefore, we restrict our investigation to *nonempty* conflict-free sets (denoted $\mathrm{CF}_{\neq\emptyset}$) and *nonempty* admissible sets (denoted $\mathrm{AD}_{\neq\emptyset}$), for which the Skeptical-Acceptance problem is nontrivial, while the Credulous-Acceptance problem remains unaffected by this change.

Secondly, if an argumentation framework *AF* has no **s** extension, then no argument is credulously accepted in *AF* with respect to **s**, but at the same time each argument is skeptically accepted. This behavior is due to the convention that a universal quantifier over an empty set (here, the set of **s** extensions) defaults to `true`, but it might be counterintuitive to call an argument skeptically accepted when it is in fact never accepted. This situation can occur for any semantics **s** that does not guarantee the existence of an extension—in our work, this is the case for the $\mathrm{CF}_{\neq\emptyset}$, $\mathrm{AD}_{\neq\emptyset}$, and ST semantics. Dunne and Wooldridge [44] propose a refined version of **s**-SA for semantics **s** that do not guarantee the existence of an extension. The refined problem additionally requires the existence of at least one **s** extension in order to give a "yes" answer. We call this problem **s**-Existence-and-Skeptical-Acceptance (**s**-ExSA), since it is the intersection of **s**-Existence (asking whether there exist an **s** extension of the given argumentation framework) and **s**-Skeptical-Acceptance.

s-Existence-and-Skeptical-Acceptance (**s**-ExSA)	
Given:	An argumentation framework $\langle \mathscr{A}, \mathscr{R} \rangle$ and an argument $a \in \mathscr{A}$.
Question:	Is there at least one **s** extension in $\langle \mathscr{A}, \mathscr{R} \rangle$ and is a in all **s** extensions of $\langle \mathscr{A}, \mathscr{R} \rangle$?

The ExSA problem can equivalently be represented as the intersection of credulous and skeptical acceptance. For any semantics **s** that guarantees the existence of at least one extension in every argumentation framework, it is apparent that **s**-ExSA and **s**-SA are equivalent. We will therefore investigate **s**-ExSA for $\mathbf{s} \in \{\mathrm{CF}_{\neq\emptyset}, \mathrm{AD}_{\neq\emptyset}, \mathrm{ST}\}$ only.

Example 4. Recall our running example argumentation framework *AF* from Figure 1. It holds that $(AF, a) \in \mathrm{CP}$-SA, since a is a member of every complete extension of *AF*. Further, we have $(AF, c) \in \mathrm{CP}$-CA, but $(AF, c) \notin \mathrm{CP}$-SA, since argument c occurs in one, but not all complete extensions of *AF*. For argument b, we have $(AF, b) \notin \mathrm{CP}$-CA (and thus $(AF, b) \notin \mathrm{CP}$-SA), since b is not a member of any complete extension.

Since *AF* has no stable extension, it holds that $(AF, arg) \in \mathrm{ST}$-SA for all arguments $arg \in \mathscr{A}$, but on the other hand, $(AF, arg) \notin \mathrm{ST}$-ExSA for any argument $arg \in \mathscr{A}$.

The following relations hold between the CA and SA problems.

Observation 5. AD-CA $=$ $\mathrm{AD}_{\neq\emptyset}$-CA $=$ CP-CA $=$ PR-CA, since for every argumentation framework, the union of all its (nonempty) admissible sets, the union of all its complete extensions, and the union of all its preferred extensions is the same.

Observation 6. GR-CA $=$ GR-SA, since every argumentation framework has a single unique grounded extension.

Observation 7. GR-SA $=$ CP-SA, since the intersection of all complete extensions in an argumentation framework is its grounded extension.

We obtain GR-CA $=$ GR-SA $=$ CP-SA from the combination of Observations 6 and 7.

2.2. Incomplete Argumentation Frameworks

An incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ splits both the set of arguments and the set of attacks into two disjoint parts, the *definite* part ($\mathscr{A}$ and $\mathscr{R}$) and the *uncertain* part ($\mathscr{A}^?$ and $\mathscr{R}^?$), where both $\mathscr{R}$ and $\mathscr{R}^?$ are subsets of $(\mathscr{A} \cup \mathscr{A}^?) \times (\mathscr{A} \cup \mathscr{A}^?)$. For uncertain elements (members of $\mathscr{A}^?$ or $\mathscr{R}^?$), it is not known whether they are part of the argumentation—they might be added or removed in the future, or the uncertainty may just represent the limited knowledge of some agent about those elements. Definite arguments (elements of $\mathscr{A}$) are known to exist,

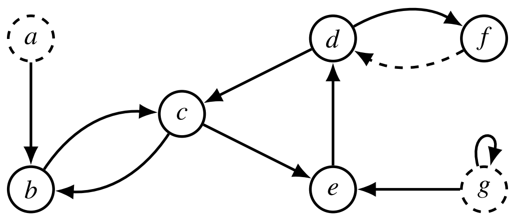


Figure 3: Graph representation of the incomplete argumentation framework in Example 8.

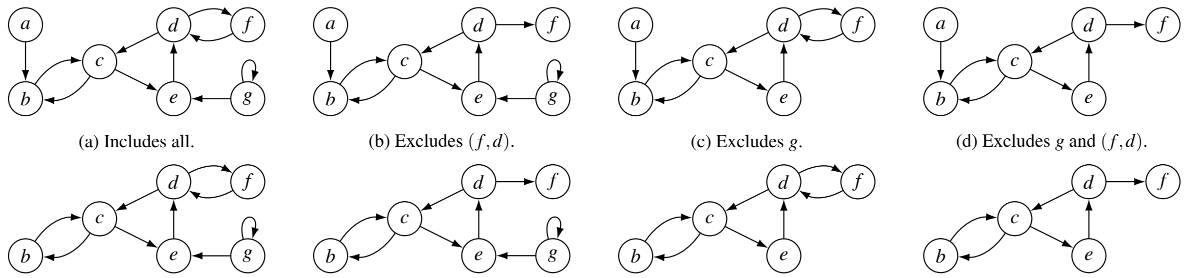

(a) Includes all. (b) Excludes (f,d). (c) Excludes g. (d) Excludes g and (f,d).
(e) Excludes a. (f) Excludes a and (f,d). (g) Excludes a and g. (h) Excludes a, g, and (f,d).

Figure 4: All completions of the incomplete argumentation framework displayed in Figure 3.

while definite attacks (elements of $\mathscr{R}$) exist if and only if both incident arguments exist, too. To account for this, we call attacks in $\mathscr{R}$ that are incident to at least one uncertain argument *conditionally definite*, since these attacks may vanish alongside an incident uncertain argument. On the other hand, attacks in $\mathscr{R}$ that are incident to only definite arguments are called *definite*. If $\mathscr{A}^? = \emptyset$, we have a *(purely) attack-incomplete* argumentation framework [33]; for $\mathscr{R}^? = \emptyset$, a *(purely) argument-incomplete* argumentation framework; and $\mathscr{A}^? = \mathscr{R}^? = \emptyset$ yields standard argumentation frameworks without uncertainty. An attack-incomplete argumentation framework may be abbreviated as $\langle \mathscr{A}, \mathscr{R}, \mathscr{R}^? \rangle$ and an argument-incomplete argumentation framework as $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R} \rangle$.

Example 8. Figure 3 shows a graph representation of an incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ with $\mathscr{A} = \{b,c,d,e,f\}$, $\mathscr{A}^? = \{a,g\}$, $\mathscr{R} = \{(a,b),(b,c),(c,b),(c,e),(d,c),(d,f),(e,d),(g,e),(g,g)\}$, and $\mathscr{R}^? = \{(f,d)\}$, where definite elements are solid (circles for arguments, and arrows for attacks) and uncertain elements are dashed. Note that some of the solid attacks are in fact conditionally definite. Unlike Baumeister et al. [17], who represent conditionally definite attacks by dotted arcs, we will display them here by solid arcs so as to not overload the figures with too much information. However, do keep in mind that all arcs incident to uncertain (dashed) nodes will vanish if, and only if, these nodes vanish in a completion (to be defined below). In this example, this refers to the conditionally definite attacks (a,b), (g,e), and (g,g).

A *completion* of an incomplete argumentation framework $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ is any argumentation framework $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ that satisfies $\mathscr{A} \subseteq \mathscr{A}^* \subseteq \mathscr{A} \cup \mathscr{A}^?$ and $\mathscr{R}|_{\mathscr{A}^*} \subseteq \mathscr{R}^* \subseteq \left(\mathscr{R} \cup \mathscr{R}^?\right)|_{\mathscr{A}^*}$. Here, the *restriction* $\mathscr{R}|_{\mathscr{A}^*}$ of an attack relation $\mathscr{R}$ to $\mathscr{A}^*$ is defined as $\mathscr{R}|_{\mathscr{A}^*} = \{(a,b) \in \mathscr{R} \mid a,b \in \mathscr{A}^*\}$. It represents the fact that attacks can only be part of a completion which includes both incident arguments. However, a conditionally definite attack must be present in all completions containing both incident arguments, while an uncertain attack may vanish in a completion that contains both of its incident arguments.

If at least one completion of an incomplete argumentation framework *IAF* satisfies some property, this property is said to hold *possibly* for *IAF*. On the other hand, if all completions of *IAF* satisfy a property, it is said to hold *necessarily* for *IAF*. Accordingly, we define both a possible and a necessary variant of the **s**-CA, **s**-SA, and **s**-ExSA problems for incomplete argumentation frameworks, for each semantics **s** considered here. The formal definitions of the **s**-NCA, **s**-PSA, and **s**-PExSA problems are as follows.

s-POSSIBLE-CREDULOUS-ACCEPTANCE (**s**-PCA)

Given:	An incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ and an argument $a \in \mathscr{A}$.
Question:	Does there exist a completion $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ of $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ and an **s** extension $\mathscr{E}$ of AF^* such that $a \in \mathscr{E}$?

s-POSSIBLE-SKEPTICAL-ACCEPTANCE (**s**-PSA)

Given:	An incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ and an argument $a \in \mathscr{A}$.
Question:	Does there exist a completion $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ of $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ such that for each **s** extension $\mathscr{E}$ of AF^*, we have $a \in \mathscr{E}$?

s-POSSIBLE-EXISTENCE-AND-SKEPTICAL-ACCEPTANCE(**s**-PEXSA)

Given:	An incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ and an argument $a \in \mathscr{A}$.
Question:	Does there exist a completion $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ of $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ such that AF^* has an **s** extension and for each **s** extension $\mathscr{E}$ of AF^*, we have $a \in \mathscr{E}$?

We define **s**-NECESSARY-CREDULOUS-ACCEPTANCE (**s**-NCA), **s**-NECESSARY-SKEPTICAL-ACCEPTANCE (**s**-NSA), and **s**-NECESSARY-EXISTENCE-AND-SKEPTICAL-ACCEPTANCE (**s**-NEXSA) analogously to **s**-PCA, **s**-PSA, and **s**-PEXSA, respectively, except that we now quantify *universally* over all completions AF^*.

The scope of the target argument a is restricted to $a \in \mathscr{A}$ instead of allowing $a \in \mathscr{A} \cup \mathscr{A}^?$ in all our problem definitions, since allowing uncertain arguments as target does not produce any additional interesting cases. Specifically, for the *necessary* problem variants, all instances with $a \in \mathscr{A}^?$ are trivial "no" instances. On the other hand, for each *possible* problem variant $\mathbf{s}\text{-}\mathscr{P} \in \{\mathbf{s}\text{-PCA}, \mathbf{s}\text{-PSA}, \mathbf{s}\text{-PExSA}\}$) and any instance $(\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle, a)$ with $a \in \mathscr{A}^?$, we have that $(\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle, a) \in \mathbf{s}\text{-}\mathscr{P}$ if and only if $(\langle \mathscr{A} \cup \{a\}, \mathscr{A}^? \setminus \{a\}, \mathscr{R}, \mathscr{R}^? \rangle, a) \in \mathbf{s}\text{-}\mathscr{P}$, so the problem instance can be reduced to an equivalent formulation where $a \in \mathscr{A}$.

Observation 9. Due to Observations 5, 6, and 7, we immediately have the following equalities:

- GR-PCA = GR-PSA = CP-PSA,
- GR-NCA = GR-NSA = CP-NSA,
- AD-PCA = $\text{AD}_{\neq\emptyset}$-PCA = CP-PCA = PR-PCA, and
- AD-NCA = $\text{AD}_{\neq\emptyset}$-NCA = CP-NCA = PR-NCA.

Example 10. All completions of the incomplete argumentation framework from Figure 3 are displayed in Figure 4. We summarize the preferred and stable extensions of all completions (completions are identified by the label of the respective subfigure):

semantics	4a	4b	4c	4d	4e	4f	4g	4h
preferred	$\{a,c,f\}$	$\{a\}$	$\{a,c,f\}$	$\{a\}$	$\{c,f\}$, $\{b,f\}$	$\{b\}$	$\{b,e,f\}$, $\{c,f\}$	$\{b,e,f\}$
stable	–	–	$\{a,c,f\}$	–	–	–	$\{b,e,f\}$, $\{c,f\}$	$\{b,e,f\}$

We make a few observations:

- $(IAF, d) \notin$ PR-PCA, since no completion has d in any of its preferred extensions.
- $(IAF, b) \in$ PR-PSA, because there are completions that have b in all of their preferred extensions. But on the other hand, $(IAF, b) \notin$ PR-NCA, since there are also completions that have b in none of their preferred extensions.
- $(IAF, f) \in$ ST-NSA, because every completion either has no stable extension (resulting in a trivial "yes" answer for every ST-SA query) or has f in all of its stable extensions. On the other hand, $(IAF, f) \notin$ ST-NEXSA, since not every completion has a stable extension.

Table 1: Overview of previously known and new complexity results for credulous and skeptical acceptance problems, where previously known results are ascribed to the respective source (references are numbers in brackets), and new results are labeled with the respective theorem number. Results marked with an asterisk (*) are straight-forward. Results for EXSA problem variants (in parentheses) are given only when they differ from the corresponding result for SA. For any complexity class $\mathscr{C}$, $\mathscr{C}$-c. stands as a shorthand for "$\mathscr{C}$-complete." Furthermore, "trivial" means that the answer is always "no," while "$\in$ P" implies the existence of a linear-time or constant-time algorithm.

s	**s**-CA		**s**-PCA		**s**-NCA		**s**-SA (**s**-EXSA)		**s**-PSA (**s**-PEXSA)		**s**-NSA (**s**-NEXSA)	
CF	$\in$ P	11	$\in$ P	11	$\in$ P	11	trivial	*	trivial	*	trivial	*
CF$_{\neq\emptyset}$							$\in$ P	11	$\in$ P	11	$\in$ P	11
AD	NP-c.	[38]	NP-c.	13	Π_2^p-c.	24	trivial	*	trivial	*	trivial	*
AD$_{\neq\emptyset}$							coNP-c. (DP-c.)	33 (40)	Σ_2^p-c.	33	coNP-c. (Π_2^p-c.)	13 (36)
ST	NP-c.	[38]	NP-c.	13	Π_2^p-c.	24	coNP-c. (DP-c.)	[38] ([44])	Σ_2^p-c.	25	coNP-c. (Π_2^p-c.)	13 (36)
CP	NP-c.	[34]	NP-c.	13	Π_2^p-c.	24	P-c.	[34, 49]	NP-c.	21	coNP-c.	23
GR	P-c.	[49]	NP-c.	20	coNP-c.	22	P-c.	[49]	NP-c.	21	coNP-c.	23
PR	NP-c.	[38]	NP-c.	13	Π_2^p-c.	24	Π_2^p-c.	[43]	Σ_3^p-c.	29	Π_2^p-c.	13

3. Complexity Results

In this section, we will fully characterize the computational complexity of all variants of the CA and SA problems defined above for incomplete argumentation frameworks. Our results (together with previously known results) are summarized in Table 1.

We assume the reader to be familiar with the needed notions from computational complexity theory, such as the concepts of hardness and completeness based on polynomial-time many-one reducibility as well as the classes of the polynomial hierarchy due to Meyer and Stockmeyer [65, 80] and of the boolean hierarchy studied early on by Cai et al. [25] and Köbler et al. [57]. In particular, we will consider the well-known classes capturing deterministic and nondeterministic polynomial time, P and NP, the class DP due to Papadimitriou and Yannakakis [76] (which belongs to the second level of the boolean hierarchy and has the canonical complete problem 3-SAT-UNSAT, defined as the intersection of 3-SAT and 3-UNSAT; for the latter two problems, see Table 2 on page 12), and the classes Σ_2^p, Π_2^p, and Σ_3^p from the second and third levels of the polynomial hierarchy with their canonical complete problems Σ_2SAT, Π_2SAT, and Σ_3SAT (again, see Table 2 on page 12 for their definitions). More background on computational complexity can be found, for instance, in the textbooks by Papadimitriou [75] and Rothe [78].

The computational complexity of the original CA and SA problems for conflict-freeness, admissibility, and the stable, complete, grounded, and preferred semantics ranges from P membership to Π_2^p-completeness and was investigated by Dimopoulos and Torres [38], Dunne and Bench-Capon [43], Coste-Marquis et al. [34], and Dunne and Wooldridge [44] (again, see Table 1). In the remainder of this section, we will present the proofs for our complexity results.

3.1. Upper Bounds

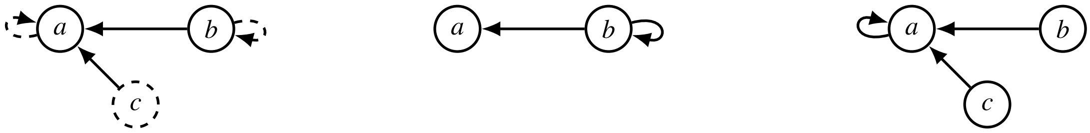
We start with P membership results for all problem variants that use the CF$_{\neq\emptyset}$ semantics.

Proposition 11. *The following problems are all in* P.

- CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, CF$_{\neq\emptyset}$-EXSA,
- CF$_{\neq\emptyset}$-PCA, CF$_{\neq\emptyset}$-PSA, CF$_{\neq\emptyset}$-PEXSA,
- CF$_{\neq\emptyset}$-NCA, CF$_{\neq\emptyset}$-NSA, CF$_{\neq\emptyset}$-NEXSA.

Proof. Whenever $(a,a) \in \mathscr{R}$, any set containing a cannot be conflict-free, and whenever $(a,a) \notin \mathscr{R}$, $\{a\}$ is a nonempty conflict-free set that contains a. We can infer that $(\langle \mathscr{A}, \mathscr{R} \rangle, a) \in$ CF$_{\neq\emptyset}$-CA if and only if $(a,a) \notin \mathscr{R}$; that $(\langle \mathscr{A}, \mathscr{R} \rangle, a) \in$ CF$_{\neq\emptyset}$-SA if and only if $(b,b) \in \mathscr{R}$ for each $b \in \mathscr{A} \setminus \{a\}$; and that $(\langle \mathscr{A}, \mathscr{R} \rangle, a) \in$ CF$_{\neq\emptyset}$-EXSA if and only if both of the above conditions hold simultaneously. All these criteria can be verified in polynomial time, so CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, and CF$_{\neq\emptyset}$-EXSA are in P.

For an instance (IAF, a) with $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ of CF$_{\neq\emptyset}$-PCA, CF$_{\neq\emptyset}$-PSA, or CF$_{\neq\emptyset}$-PEXSA, we construct a single completion $AF^{pos} = \langle \mathscr{A}^{pos}, \mathscr{R}^{pos} \rangle$ with $\mathscr{A}^{pos} = \mathscr{A}$ and $\mathscr{R}^{pos} = \left(\mathscr{R} \cup \{(b,b) \in \mathscr{R}^? \mid b \neq a\}\right)|_{\mathscr{A}^{pos}}$. An example



(a) *IAF* (b) AF^{pos} (c) AF^{nec}

Figure 5: Example for the completions AF^{pos} and AF^{nec} used in the proof of Proposition 11. Argument a is credulously and skeptically accepted in AF^{pos}, and therefore possibly credulously and possibly skeptically accepted in *IAF*. Argument a is not credulously and not skeptically accepted in AF^{nec}, and therefore not necessarily credulously and not necessarily skeptically accepted in *IAF*.

is given in Figure 5. AF^{pos} can be constructed in polynomial time and we have that (IAF,a) is a "yes" instance of CF$_{\neq\emptyset}$-PCA, CF$_{\neq\emptyset}$-PSA, and CF$_{\neq\emptyset}$-PExSA, respectively, if and only if (AF^{pos},a) is a "yes" instance of the corresponding base problem CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, and CF$_{\neq\emptyset}$-ExSA, respectively. The implication from the base problem to its *possible* generalization is trivial. For the other direction of the equivalence, we cover the problems CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, and CF$_{\neq\emptyset}$-ExSA individually.

- If $(AF^{pos},a) \notin$ CF$_{\neq\emptyset}$-CA, then there is no conflict-free set in AF^{pos} that contains a. This can only be the case if $(a,a) \in \mathscr{R}^{pos}$, which by construction of $\mathscr{R}^{pos}$ means that $(a,a) \in \mathscr{R}$ is a definite attack. This implies that a cannot be in a conflict-free set in any completion, so $(IAF,a) \notin$ CF$_{\neq\emptyset}$-PCA.
- If $(AF^{pos},a) \notin$ CF$_{\neq\emptyset}$-SA, then there is a nonempty conflict-free set $\mathscr{E} \subseteq \mathscr{A}^{pos}$ that does not contain a. This can only be the case if $(b,b) \notin \mathscr{R}^{pos}$ for some $b \neq a$ in $\mathscr{A}^{pos}$. Since all possible self-attacks of arguments other than a are included in $\mathscr{R}^{pos}$ and since $b \in \mathscr{A}^{pos} = \mathscr{A}$ is a definite argument, this means that $b \in \mathscr{A}^*$ and $(b,b) \notin \mathscr{R}^*$ for any completion $\langle \mathscr{A}^*, \mathscr{R}^* \rangle$. Therefore, $\{b\}$ is a conflict-free set not containing a in every completion, so $(IAF,a) \notin$ CF$_{\neq\emptyset}$-PSA.
- If $(AF^{pos},a) \notin$ CF$_{\neq\emptyset}$-ExSA, then there is no conflict-free set in AF^{pos} that contains a, or there is a nonempty conflict-free set $\mathscr{E} \subseteq \mathscr{A}^{pos}$ that does not contain a. The previous proofs provide that $(IAF,a) \notin$ CF$_{\neq\emptyset}$-PCA or that $(IAF,a) \notin$ CF$_{\neq\emptyset}$-PSA, and therefore (IAF,a) cannot be in the intersection CF$_{\neq\emptyset}$-PExSA of these problems.

Analogously, for an instance (IAF,a) with $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ of CF$_{\neq\emptyset}$-NCA, CF$_{\neq\emptyset}$-NSA, or CF$_{\neq\emptyset}$-NExSA, we construct a single completion $AF^{nec} = \langle \mathscr{A}^{nec}, \mathscr{R}^{nec} \rangle$ with $\mathscr{A}^{nec} = \mathscr{A} \cup \mathscr{A}^?$ and $\mathscr{R}^{nec} = \mathscr{R} \cup \big(\{(a,a)\} \cap \mathscr{R}^?\big)$. Again, an example is given in Figure 5. AF^{nec} can be constructed in polynomial time and we have that (IAF,a) is a "yes" instance of CF$_{\neq\emptyset}$-NCA, CF$_{\neq\emptyset}$-NSA, and CF$_{\neq\emptyset}$-NExSA, respectively, if and only if (AF^{nec},a) is a "yes" instance of the corresponding base problem CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, and CF$_{\neq\emptyset}$-ExSA, respectively. Here, the implication from the *necessary* generalization to the base problem is trivial. For the other direction of the equivalence, we again cover the problems CF$_{\neq\emptyset}$-CA, CF$_{\neq\emptyset}$-SA, and CF$_{\neq\emptyset}$-ExSA individually.

- If $(AF^{nec},a) \in$ CF$_{\neq\emptyset}$-CA, then there is a nonempty conflict-free set $\mathscr{E} \subseteq \mathscr{A}^{nec}$ with $a \in \mathscr{E}$. This can only be the case if $(a,a) \notin \mathscr{R}^{nec}$. Due to the construction of AF^{nec}, we know that $(a,a) \notin \mathscr{R}^*$ for any completion $\langle \mathscr{A}^*, \mathscr{R}^* \rangle$, and since a is definite, this means that $\{a\}$ is a nonempty conflict-free set in every completion, so $(IAF,a) \in$ CF$_{\neq\emptyset}$-NCA.
- If $(AF^{nec},a) \in$ CF$_{\neq\emptyset}$-SA, then there is no nonempty conflict-free set $\mathscr{E} \subseteq \mathscr{A}^{nec}$ with $a \notin \mathscr{E}$. This can only be the case if $(b,b) \in \mathscr{R}^{nec}$ for all arguments $b \in \mathscr{A}^{nec}$ with $b \neq a$. Due to the fact that $\mathscr{A}^{nec} = \mathscr{A} \cup \mathscr{A}^?$ contains all possible arguments, and because $\mathscr{R}^{nec}$ contains only definite self-attacks for arguments $b \neq a$, we can conclude that $(b,b) \in \mathscr{R}^*$ for all arguments $b \in \mathscr{A}^*$ with $b \neq a$ holds for every completion $\langle \mathscr{A}^*, \mathscr{R}^* \rangle$. Thus there can be no nonempty conflict-free set without a in any completion, and therefore $(IAF,a) \in$ CF$_{\neq\emptyset}$-NSA.
- If $(AF^{nec},a) \in$ CF$_{\neq\emptyset}$-ExSA, then $(AF^{nec},a) \in ($CF$_{\neq\emptyset}$-CA $\cap$ CF$_{\neq\emptyset}$-SA$)$, and by the previous arguments, $(IAF,a) \in$ CF$_{\neq\emptyset}$-NCA and $(IAF,a) \in$ CF$_{\neq\emptyset}$-NSA, so $(IAF,a) \in$ CF$_{\neq\emptyset}$-NExSA.

This concludes the proof. ❑

In Proposition 12, we derive upper bounds for all remaining acceptance problem variants from their respective quantifier representations. In Section 3.2, we will prove matching lower bounds.

Proposition 12. *The following complexity upper bounds hold.*

1. *For* $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}, \text{CP}, \text{GR}, \text{PR}\}$, $\mathbf{s}$-PCA *is in* NP*, and for* $\mathbf{s}' \in \{\text{CP}, \text{GR}\}$, $\mathbf{s}'$-PSA *is in* NP.
2. $\text{AD}_{\neq\emptyset}$-SA*,* GR-NCA*, and* $\mathbf{s}$-NSA *for* $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}, \text{CP}, \text{GR}\}$ *are in* coNP.
3. $\text{AD}_{\neq\emptyset}$-ExSA *is in* DP.
4. PR-NSA *is in* Π_2^p, $\mathbf{s}$-NCA *is in* Π_2^p *for* $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}, \text{CP}, \text{PR}\}$*, and* $\mathbf{s}$'-NExSA *is in* Π_2^p *for* $\mathbf{s}' \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$.
5. *For* $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, $\mathbf{s}$-PSA *and* $\mathbf{s}$-PExSA *are in* Σ_2^p.
6. PR-PSA *is in* Σ_3^p.

Proof. All acceptance problem variants can be represented using a sequence of polynomially length-bounded existential or universal quantifiers (which we will simply call "existential quantifiers" or "universal quantifiers" further on) followed by a predicate that corresponds to the $\mathbf{s}$-VERIFICATION ($\mathbf{s}$-VER) problem for the respective semantics. An instance $(AF, \mathscr{E})$ of $\mathbf{s}$-VER—where $AF = \langle \mathscr{A}, \mathscr{R} \rangle$ is an argumentation framework and $\mathscr{E} \subseteq \mathscr{A}$ is a set of arguments in AF—is a "yes" instance if and only if $\mathscr{E}$ is an $\mathbf{s}$ extension of AF. It is known [41] that PR-VER is coNP-complete and $\mathbf{s}$-VER is in P for all other semantics used in the current article (clearly, the complexity of $\text{CF}_{\neq\emptyset}$-VER and CF-VER is the same, as is that of $\text{AD}_{\neq\emptyset}$-VER and AD-VER).

The quantifier representations for $\mathbf{s}$-CA and $\mathbf{s}$-SA are as follows:

$$(\langle \mathscr{A}, \mathscr{R} \rangle, a) \in \mathbf{s}\text{-CA} \quad \Leftrightarrow \quad (\exists \mathscr{E} \subseteq \mathscr{A} \setminus \{a\}) \left[(\langle \mathscr{A}, \mathscr{R} \rangle, \mathscr{E} \cup \{a\}) \in \mathbf{s}\text{-VER} \right];$$
$$(\langle \mathscr{A}, \mathscr{R} \rangle, a) \in \mathbf{s}\text{-SA} \quad \Leftrightarrow \quad (\forall \mathscr{E} \subseteq \mathscr{A} \setminus \{a\}) \left[(\langle \mathscr{A}, \mathscr{R} \rangle, \mathscr{E}) \notin \mathbf{s}\text{-VER} \right].$$

For all semantics $\mathbf{s}$ for which $\mathbf{s}$-VER is in P, this representation immediately provides an NP upper bound for $\mathbf{s}$-CA and a coNP upper bound for $\mathbf{s}$-SA. In particular, we have $\text{AD}_{\neq\emptyset}\text{-SA} \in \text{coNP}$. The $\mathbf{s}$-ExSA problem is the intersection of $\mathbf{s}$-CA and $\mathbf{s}$-SA and thus has a DP upper bound. In particular, $\text{AD}_{\neq\emptyset}$-ExSA is in DP.

The *possible* (respectively, *necessary*) generalization of the acceptance problems for incomplete argumentation frameworks adds an existential (respectively, universal) quantifier over completions to the representation:

$$(\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle, a) \in \mathbf{s}\text{-PCA} \quad \Leftrightarrow$$
$$\left(\exists \text{ completion } \langle \mathscr{A}^*, \mathscr{R}^* \rangle \text{ of } \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle\right) \left[(\langle \mathscr{A}^*, \mathscr{R}^* \rangle, a) \in \mathbf{s}\text{-CA} \right];$$
$$(\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle, a) \in \mathbf{s}\text{-NCA} \quad \Leftrightarrow$$
$$\left(\forall \text{ completion } \langle \mathscr{A}^*, \mathscr{R}^* \rangle \text{ of } \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle\right) \left[(\langle \mathscr{A}^*, \mathscr{R}^* \rangle, a) \in \mathbf{s}\text{-CA} \right].$$

The representations of $\mathbf{s}$-PSA, $\mathbf{s}$-NSA, $\mathbf{s}$-PExSA, and $\mathbf{s}$-NExSA only differ in that they use $\mathbf{s}$-SA and $\mathbf{s}$-ExSA, respectively, instead of $\mathbf{s}$-CA in the inner predicate.

For semantics $\mathbf{s}$ with $\mathbf{s}\text{-VER} \in \text{P}$, these representations allow to infer the following bounds. For $\mathbf{s}$-PSA and $\mathbf{s}$-PExSA, we get a Σ_2^p upper bound, since their quantifier representation has an existential quantifier followed by a, respectively, coNP and DP predicate. Similarly, we obtain a Π_2^p upper bound for $\mathbf{s}$-NCA and $\mathbf{s}$-NExSA, since their quantifier representation has a universal quantifier followed by, respectively, an NP and a DP predicate. Since two subsequent existential or two subsequent universal quantifiers can be collapsed into one such quantifier, we get an NP upper bound for $\mathbf{s}$-PCA and a coNP upper bound for $\mathbf{s}$-NSA. For the preferred semantics, where the inner predicate is a coNP instead of a P predicate, we obtain a Σ_2^p upper bound for PR-PCA, a Π_2^p upper bound for PR-NSA, a Σ_3^p upper bound for PR-PSA, and a Π_3^p upper bound for PR-NCA.

Of these upper bounds, not all will turn out to be tight. We utilize the NP membership of $\text{AD}_{\neq\emptyset}$-PCA, ST-PCA, CP-PCA, and GR-PCA; the coNP membership of $\text{AD}_{\neq\emptyset}$-NSA, ST-NSA, CP-NSA, and GR-NSA; the Σ_2^p membership of $\text{AD}_{\neq\emptyset}$-PSA, ST-PSA, $\text{AD}_{\neq\emptyset}$-PExSA, and ST-PExSA; the Π_2^p membership of $\text{AD}_{\neq\emptyset}$-NCA, ST-NCA, CP-NCA, $\text{AD}_{\neq\emptyset}$-NExSA, ST-NExSA, and PR-NSA; and the Σ_3^p membership of PR-PSA.

Further, PR-PCA inherits the NP upper bound of $\text{AD}_{\neq\emptyset}$-PCA and PR-NCA inherits the Π_2^p upper bound of $\text{AD}_{\neq\emptyset}$-NCA via Observation 5; GR-PSA inherits the NP upper bound of GR-PCA and GR-NCA inherits the coNP upper bound of GR-NSA via Observation 6; and CP-PSA inherits the NP upper bound of GR-PSA via Observation 7.

This completes the proof. ❑

Table 2: Overview of different QSAT problems and their complexities.

Problem	Instance	Question	Complexity
3-SAT	(φ, X)	$(\exists\tau_X)\,[\varphi[\tau_X] = \texttt{true}]$?	NP-complete
3-UNSAT	(φ, X)	$(\forall\tau_X)\,[\varphi[\tau_X] = \texttt{false}]$?	coNP-complete
Σ_2SAT	(φ, X, Y)	$(\exists\tau_X)(\forall\tau_Y)\,[\varphi[\tau_X, \tau_Y] = \texttt{false}]$?	Σ_2^p-complete
Π_2SAT	(φ, X, Y)	$(\forall\tau_X)(\exists\tau_Y)\,[\varphi[\tau_X, \tau_Y] = \texttt{true}]$?	Π_2^p-complete
Σ_3SAT	(φ, X, Y, Z)	$(\exists\tau_X)(\forall\tau_Y)(\exists\tau_Z)\,[\varphi[\tau_X, \tau_Y, \tau_Z] = \texttt{true}]$?	Σ_3^p-complete

3.2. Lower Bounds

The *possible* and *necessary* variants of acceptance problems are true generalizations of their base problem. Whenever the uncertain elements in an instance are empty (i.e., $\mathscr{A}^? = \emptyset$ and $\mathscr{R}^? = \emptyset$), both variants collapse to their base problem. Therefore, the *possible* and *necessary* variants inherit all lower complexity bounds from their base problems. In some cases, these bounds match the corresponding upper bounds from Section 3.1 and provide tight characterizations. We collect these cases in Corollary 13.

Corollary 13. *1. For* $\mathbf{s} \in \{\mathrm{AD}_{\neq\emptyset}, \mathrm{ST}, \mathrm{CP}, \mathrm{PR}\}$*,* $\mathbf{s}$-PCA *is* NP*-hard.*

2. For $\mathbf{s} \in \{\mathrm{AD}_{\neq\emptyset}, \mathrm{ST}\}$*,* $\mathbf{s}$-NSA *is* coNP*-hard.*

3. PR-NSA *is* Π_2^p*-hard.*

For all remaining cases, we obtain tight lower bounds by reducing from different quantified satisfiability (QSAT) problems. The QSAT problems that we require are defined in Table 2. We use X, Y, and Z to denote pairwise disjoint sets of propositional variables, and φ to represent a boolean formula in conjunctive normal form—i.e., a conjunction of disjunctive clauses of literals—with at most three literals per clause (3-CNF). τ_X with $\tau_X : X \to \{\texttt{true}, \texttt{false}\}$ denotes a truth assignment on a set X of propositional variables, and for a formula φ over X, $\varphi[\tau_X]$ denotes the truth value that φ evaluates to when applying assignment τ_X to φ.

Each QSAT problem variant is complete for a different class in the polynomial hierarchy (last column of Table 2). We use a translation from QSAT instances to instances of acceptance problems for incomplete argumentation frameworks that is based on the standard translation by Dimopoulos and Torres [38], which we extend to incorporate uncertainty about arguments or attacks. Our translation is given in Definition 14. It allows to create either a purely argument-incomplete argumentation framework (where $\mathscr{R}^? = \emptyset$) or a purely attack-incomplete argumentation framework (where $\mathscr{A}^? = \emptyset$), which allows the hardness results obtained to hold even in these special cases.

Definition 14. Let (φ, X, Y) be an instance of Σ_2SAT or Π_2SAT, and let (φ, X) (with $Y = \emptyset$) be an instance of 3-SAT or 3-UNSAT. Let $\varphi = \bigwedge_i c_i$ and $c_i = \bigvee_j \alpha_j$ for each clause c_i in φ, where the α_j are the literals over $X \cup Y$ that occur in clause c_i. We define a set of arguments $\mathscr{A}$ and a set of attacks $\mathscr{R}$ as follows:

$$\begin{aligned}
\mathscr{A} =& \{x_i, \bar{x}_i \mid x_i \in X\} \cup \{y_i, \bar{y}_i \mid y_i \in Y\} \cup \{\bar{c}_i \mid c_i \text{ in } \varphi\} \cup \{\varphi, \bar{\varphi}\};\\
\mathscr{R} =& \{(\bar{x}_i, x_i) \mid x_i \in X\} \cup \{(y_i, \bar{y}_i), (\bar{y}_i, y_i) \mid y_i \in Y\}\\
&\cup \{(x_k, \bar{c}_i) \mid x_k \text{ in } c_i\} \cup \{(\bar{x}_k, \bar{c}_i) \mid \neg x_k \text{ in } c_i\}\\
&\cup \{(y_k, \bar{c}_i) \mid y_k \text{ in } c_i\} \cup \{(\bar{y}_k, \bar{c}_i) \mid \neg y_k \text{ in } c_i\}\\
&\cup \{(\bar{c}_i, \varphi) \mid c_i \in \varphi\} \cup \{(\varphi, \bar{\varphi})\}.
\end{aligned}$$

An attack-incomplete argumentation framework representation of the given QSAT instance is given by $\langle \mathscr{A} \cup \{g_i \mid x_i \in X\}, \emptyset, \mathscr{R}, \mathscr{R}^? \rangle$ with $\mathscr{R}^? = \{(g_i, \bar{x}_i) \mid x_i \in X\}$. Alternatively, an argument-incomplete argumentation framework representation of the given QSAT instance is given by $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R} \cup \{(g_i, \bar{x}_i) \mid x_i \in X\}, \emptyset \rangle$ with $\mathscr{A}^? = \{g_i \mid x_i \in X\}$.

We call arguments x_i, y_i, $\bar{x}_i$, and $\bar{y}_i$ *literal arguments* and arguments $\bar{c}_i$ *clause arguments*. We call two arguments $\bar{a}$ and a *counterparts* of each other. The unattacked arguments g_i are the *grounded* arguments which enforce the acceptability of either x_i or $\bar{x}_i$ via the completion. Arguments φ and $\bar{\varphi}$ will be used as the target arguments in acceptance problems and are called *target* arguments.

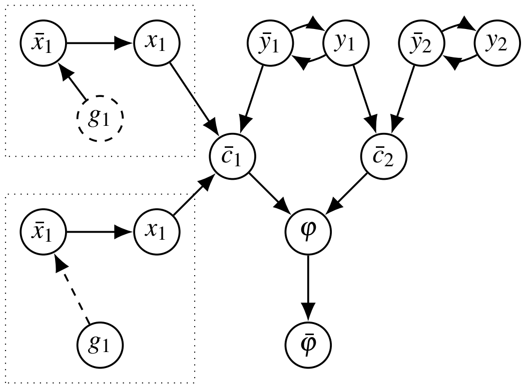


Figure 6: Visualization of either the argument-incomplete argumentation framework (not including the bottom framed part) or attack-incomplete argumentation framework (not including the top framed part) created by Definition 14 for the clauses $c_1 = x_1 \vee \neg y_1$ and $c_2 = y_1 \vee \neg y_2$ given in Example 15.

Example 15. Let (φ, X, Y) be a QSAT instance with $X = \{x_1\}$, $Y = \{y_1, y_2\}$, $\varphi = c_1 \wedge c_2$, $c_1 = x_1 \vee \neg y_1$, and $c_2 = y_1 \vee \neg y_2$. Both the argument-incomplete argumentation framework and the attack-incomplete argumentation framework created for (φ, X, Y) by the translation of Definition 14 are displayed in Figure 6.

For an incomplete argumentation framework *IAF* created according to Definition 14, we associate a given truth assignment τ_X on X with a completion $AF^{\tau_X} = \langle \mathscr{A}^{\tau_X}, \mathscr{R}^{\tau_X} \rangle$ of *IAF*. For an attack-incomplete argumentation framework $\langle \mathscr{A}, \mathscr{R}, \mathscr{R}^? \rangle$, that completion has $\mathscr{A}^{\tau_X} = \mathscr{A}$ and $(g_i, \bar{x}_i) \in \mathscr{R}^{\tau_X} \iff \tau_X(x_i) = \texttt{true}$. For an argument-incomplete argumentation framework $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R} \rangle$, that completion has $g_i \in \mathscr{A}^{\tau_X} \iff \tau_X(x_i) = \texttt{true}$ and $\mathscr{R}^{\tau_X} = \mathscr{R}|_{\mathscr{A}^{\tau_X}}$. Further, we identify an assignment τ_S on a set $S = \{s_1, \ldots, s_{|S|}\} \subseteq (X \cup Y)$ of variables with a set $\mathscr{A}^{\tau_X}[\tau_S]$ of arguments in the completion, namely, $\mathscr{A}^{\tau_X}[\tau_S] = \{s_i \mid \tau_S(s_i) = \texttt{true}\} \cup \{\bar{s}_i \mid \tau_S(s_i) = \texttt{false}\} \cup G[\tau_S]$, where $G[\tau_S] = \{g_i \mid x_i \in X \text{ and } \tau_S(x_i) = \texttt{true}\}$ for the argument-incomplete encoding, and $G[\tau_S] = \{g_i \mid x_i \in X\}$ for the attack-incomplete encoding.

In Lemma 16, we prove that both constructions behave similarly and can, in effect, be used interchangeably.

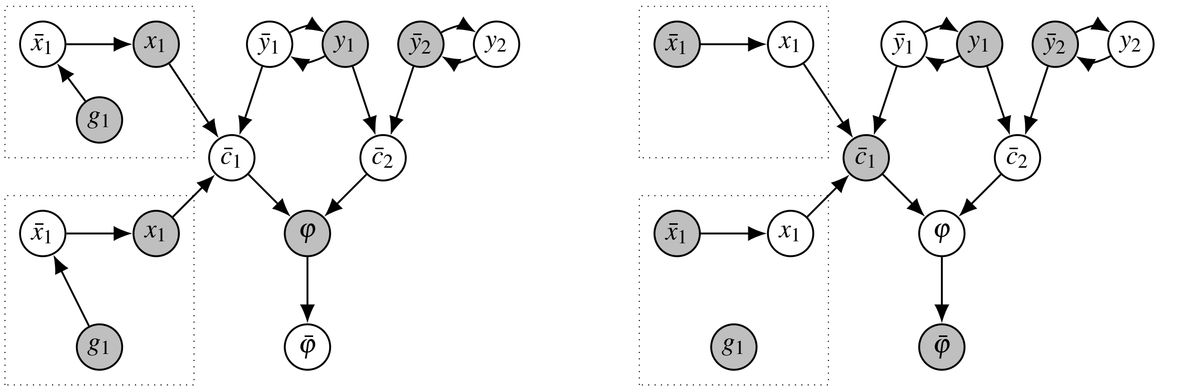
Lemma 16. *Let (φ, X, Y) be a* QSAT *instance, let $\langle \mathscr{A}, \mathscr{R}, \mathscr{R}^? \rangle$ or $\langle \mathscr{A}, \mathscr{A}^?, \mathscr{R} \rangle$ be an incomplete argumentation framework created for it according to Definition 14, and let τ_X be an assignment on X. In the completion AF^{τ_X}, $\mathscr{A}^{\tau_X}[\tau_X]$ is a subset of the grounded extension and therefore a subset of all complete extensions.*

Proof. Each argument g_i is always unattacked and therefore clearly in the grounded extension of every completion that contains g_i. Each argument $\bar{x}_i$ with $\tau_X(x_i) = \texttt{true}$ is attacked by the corresponding argument g_i, which thus defends the counterpart x_i of $\bar{x}_i$, so these x_i are included in the grounded extension. All arguments $\bar{x}_j$ for which $\tau_X(x_j) = \texttt{false}$ remain unattacked, since either $g_i \notin \mathscr{A}^{\tau_X}$ (argument-incomplete variant) or $(g_i, \bar{x}_i) \notin \mathscr{R}^{\tau_X}$ (attack-incomplete variant), and are thus included in the grounded extension. ❑

Example 17. Recall the QSAT instance from Example 15. The set $X = \{x_1\}$ allows two assignments, τ_X^1 with $\tau_X^1(x_1) = \texttt{true}$ and τ_X^2 with $\tau_X^2(x_1) = \texttt{false}$. The completion $AF^{\tau_X^1}$ of *IAF* includes the uncertain argument g_1 (in case of the argument-incomplete version) or the uncertain attack $(g_1, \bar{x}_1)$ (in case of the attack-incomplete version), while the completion $AF^{\tau_X^2}$ excludes g_1 or $(g_1, \bar{x}_1)$, respectively. In both incarnations of $AF^{\tau_X^1}$, the grounded extension is $\{g_1, x_1\} = \mathscr{A}^{\tau_X^1}[\tau_X^1]$, while in $AF^{\tau_X^2}$, the grounded extension is $\{g_1, \bar{x}_1\} = \mathscr{A}^{\tau_X^2}[\tau_X^2]$ for the attack-incomplete version or $\{\bar{x}_1\} = \mathscr{A}^{\tau_X^2}[\tau_X^2]$ for the argument-incomplete version.

Lemma 18 shows a crucial correspondence between assignments in a QSAT instance and sets of arguments in the respective incomplete argumentation framework.

Lemma 18. *Given a* QSAT *instance (φ, X, Y) and full assignments τ_X and τ_Y (τ_Y only if applicable). Let IAF be an incomplete AF created for (φ, X, Y) following Definition 14, let AF^{τ_X} be its completion corresponding to τ_X, and let $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y]$ be the set of literal arguments and grounded arguments corresponding to the total assignment.*



(a) Completion $AF^{\tau_X^1}$, where the set $\mathscr{A}^{\tau_X^1}[\tau_X^1, \tau_Y] \cup \{\varphi\}$ (highlighted) is admissible, complete, preferred, and stable.

(b) Completion $AF^{\tau_X^2}$, where the set $\mathscr{A}^{\tau_X^2}[\tau_X^2, \tau_Y] \cup \{\bar{\varphi}, \bar{c}_1\}$ (highlighted) is admissible, complete, preferred, and stable.

Figure 7: Visualization of the correspondence between truth assignments and completions plus extensions for the formula $\varphi = (x_1 \vee \neg y_1) \wedge (y_1 \vee \neg y_2)$ and the assignments τ_X^1, τ_X^2, and τ_Y given in Example 15.

1. *If $\varphi[\tau_X, \tau_Y] = \texttt{true}$, then $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is admissible, complete, preferred, and stable in AF^{τ_X}, and for $Y = \emptyset$ also grounded.*

2. *If $\varphi[\tau_X, \tau_Y] = \texttt{false}$, then $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\bar{\varphi}\} \cup \{\bar{c}_i \mid \nexists d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] : (d, \bar{c}_i) \in \mathscr{R}^{\tau_X}\}$ is admissible, complete, preferred, and stable in AF^{τ_X}, and for $Y = \emptyset$ also grounded.*

Proof. Assume that $\varphi[\tau_X, \tau_Y] = \texttt{true}$ (Part 1). We know that $\mathscr{A}^{\tau_X}[\tau_X]$ is a subset of the grounded extension of AF^{τ_X}. We show that $\mathscr{E} = \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is stable in AF^{τ_X}. It is easy to see from Definition 14 that $\mathscr{E}$ is conflict-free, since there are no attacks between literal arguments for distinct variables, φ, or the g_i. Further, $\mathscr{E}$ attacks each argument in $\mathscr{A}^{\tau_X} \setminus \mathscr{E}$. Argument $\bar{\varphi}$ is attacked by $\varphi \in \mathscr{E}$. Each literal argument from X that does not occur in $\mathscr{E}$ is either excluded from the completion, attacked by its counterpart, or attacked by some g_i in $\mathscr{E}$. Each literal argument from Y that is not in $\mathscr{E}$ is attacked by its counterpart in $\mathscr{E}$. For each clause argument $\bar{c}_i$, we know by assumption that the corresponding clause c_i in φ is satisfied by the total assignment, since $\varphi[\tau_X, \tau_Y] = \texttt{true}$. Since c_i is satisfied, at least one literal in c_i must be satisfied. By construction of $\mathscr{E}$ we know that at least one literal argument corresponding to a literal in c_i is in $\mathscr{E}$, and by construction of *IAF*, this argument attacks the clause argument $\bar{c}_i$. In total, this means that all clause arguments are attacked by $\mathscr{E}$, and we proved that $\mathscr{E}$ is stable in AF^{τ_X}. Since $\mathscr{E}$ is stable, it is also preferred, complete, and admissible. For $Y = \emptyset$, the set $\mathscr{A}^{\tau_X}[\tau_X]$, which is a subset of the grounded extension by Lemma 16, already attacks all clause arguments and thus defends φ, so $\mathscr{A}^{\tau_X}[\tau_X] \cup \{\varphi\}$ is the grounded extension of AF^{τ_X}.

Now assume that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ (Part 2). Let $\mathscr{E}' = \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\bar{\varphi}\} \cup \{\bar{c}_i \mid \nexists d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] : (d, \bar{c}_i) \in \mathscr{R}^{\tau_X}\}$. First, let us show that the subset $C = \{\bar{c}_i \mid \nexists d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] : (d, \bar{c}_i) \in \mathscr{R}^{\tau_X}\}$ of $\mathscr{E}'$ is nonempty. Since $\varphi[\tau_X, \tau_Y] = \texttt{false}$, there is at least one clause c_i' in φ that is not satisfied by the total assignment, so none of the literals in c_i' is satisfied. These literals correspond to literal arguments in *IAF*, which are the only arguments in *IAF* that attack the clause argument $\bar{c}_i'$. By construction of $\mathscr{E}'$, we know that none of these arguments are in $\mathscr{E}'$, so $\mathscr{E}'$ does not attack $\bar{c}_i'$ and thus $\bar{c}_i' \in C$. We now show that $\mathscr{E}'$ is stable in AF^{τ_X}. Again, $\mathscr{E}'$ is clearly conflict-free. All literal arguments from X that do not occur in $\mathscr{E}'$ are again either excluded from the completion, attacked by some g_i, or attacked by their counterpart in $\mathscr{E}'$. Each literal argument from Y that is not in $\mathscr{E}'$ is attacked by its counterpart in $\mathscr{E}'$. Each clause argument that is not in C is attacked by some $d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y]$ due to the definition of C. Finally, argument φ is attacked by all arguments in $C \subseteq \mathscr{E}'$, of which there is at least one since $C \neq \emptyset$. Since $\mathscr{E}'$ is stable, it is also preferred, complete, and admissible. For $Y = \emptyset$, the set $\mathscr{A}^{\tau_X}[\tau_X]$, which is a subset of the grounded extension due to Lemma 16, already attacks all clause arguments in $\mathscr{A}^{\tau_X} \setminus C$ and thus defends all arguments in C, which in turn defend $\bar{\varphi}$, so $\mathscr{E}'$ is the grounded extension of AF^{τ_X}. ❑

Example 19. Recall the QSAT instance from Examples 15 and 17. Consider τ_X^1 with $\tau_X^1(x_1) = \texttt{true}$ and τ_Y with $\tau_Y(y_1) = \texttt{true}$ and $\tau_Y(y_2) = \texttt{false}$. It holds that $\varphi[\tau_X^1, \tau_Y] = \texttt{true}$. In the completion $AF^{\tau_X^1}$, the set of arguments $\mathscr{A}^{\tau_X^1}[\tau_X^1, \tau_Y]$ corresponding to these assignments is $\{x_1, y_1, \bar{y}_2, g_1\}$. By Lemma 18, we know that $\mathscr{A}^{\tau_X^1}[\tau_X^1, \tau_Y] \cup \{\varphi\}$ is admissible, complete, preferred, and stable in $AF^{\tau_X^1}$. The completion and this set of arguments are displayed in Figure 7a.

If we use τ_X^2 with $\tau_X^2(x_1) = \texttt{false}$ instead, we get $\varphi[\tau_X^2, \tau_Y] = \texttt{false}$, since clause c_1 is no longer satisfied. In the completion $AF^{\tau_X^2}$, the set of arguments $\mathscr{A}^{\tau_X^2}[\tau_X^2, \tau_Y]$ corresponding to the assignments τ_X^2 and τ_Y is $\{\bar{x}_1, y_1, \bar{y}_2, g_1\}$ for the attack-incomplete encoding and $\{\bar{x}_1, y_1, \bar{y}_2\}$ for the argument-incomplete encoding. By Lemma 18, we know that $\mathscr{A}^{\tau_X^2}[\tau_X^2, \tau_Y] \cup \{\bar{\varphi}, \bar{c}_1\}$ is admissible, complete, preferred, and stable in $AF^{\tau_X^2}$. The completion and this set of arguments are displayed in Figure 7b.

We have now completed our preparations for the hardness proofs of the remaining problems.

Theorem 20. GR-PCA *is* NP-*hard.*

Proof. We reduce from 3-SAT. Let (φ, X) be a 3-SAT instance.

If $(\varphi, X) \in$ 3-SAT, we have that $(\exists \tau_X)[\varphi[\tau_X] = \texttt{true}]$, so by Lemma 18 there exists a completion of the corresponding incomplete argumentation framework *IAF* where φ is in the grounded extension, and we have $(IAF, \varphi) \in$ GR-PCA.

If $(\varphi, X) \notin$ 3-SAT, we have that $(\forall \tau_X)[\varphi[\tau_X] = \texttt{false}]$, so $\bar{\varphi}$ is in the grounded extension of all completions of the corresponding incomplete argumentation framework *IAF*, so φ cannot be in the grounded extension of any completion, and we have $(IAF, \varphi) \notin$ GR-PCA. ❑

Together with Observation 9, the following corollary follows immediately.

Corollary 21. GR-PSA *and* CP-PSA *are* NP-*hard.*

Theorem 22. GR-NCA *is* coNP-*hard.*

Proof. We reduce from 3-UNSAT. Let (φ, X) be a 3-UNSAT instance.

If $(\varphi, X) \in$ 3-UNSAT, we have that $(\forall \tau_X)[\varphi[\tau_X] = \texttt{false}]$, so by Lemma 18, $\bar{\varphi}$ is in the grounded extension of all completions of the corresponding incomplete argumentation framework *IAF* and we have $(IAF, \bar{\varphi}) \in$ GR-NCA.

If $(\varphi, X) \notin$ 3-UNSAT, we have that $(\exists \tau_X)[\varphi[\tau_X] = \texttt{true}]$, so there exists a completion of the corresponding incomplete argumentation framework *IAF* where φ is in the grounded extension, so $\bar{\varphi}$ cannot be in the grounded extensions of all completions, and we have $(IAF, \bar{\varphi}) \notin$ GR-NCA. ❑

Again, Observation 9 immediately gives the following corollary.

Corollary 23. GR-NSA *and* CP-NSA *are* coNP-*hard.*

Theorem 24. *For* $\mathbf{s} \in \{\mathrm{AD}_{\neq\emptyset}, \mathrm{CP}, \mathrm{ST}, \mathrm{PR}\}$, $\mathbf{s}$-NCA *is* Π_2^p-*hard.*

Proof. We reduce from Π_2SAT. Let (φ, X, Y) be a Π_2SAT instance.

If $(\varphi, X, Y) \in \Pi_2$SAT, we have that $(\forall \tau_X)(\exists \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{true}]$, so by Lemma 18, for all completions of the corresponding incomplete argumentation framework *IAF*, there is a τ_Y such that the set $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is admissible, complete, preferred, and stable, so $(IAF, \varphi) \in \mathbf{s}$-NCA for $\mathbf{s} \in \{\mathrm{AD}_{\neq\emptyset}, \mathrm{CP}, \mathrm{ST}, \mathrm{PR}\}$.

If $(\varphi, X, Y) \notin \Pi_2$SAT, we have that $(\exists \tau_X)(\forall \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{false}]$, so there is a completion AF^{τ_X} of the corresponding incomplete argumentation framework *IAF* where $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\bar{\varphi}\} \cup \{c_i \mid \nexists d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] : (d, c_i) \in \mathscr{R}^{\tau_X}\}$ is stable for any choice of τ_Y. This means that φ cannot be a member of any admissible set in that completion—and therefore neither in a complete, stable, or preferred set—so $(IAF, \varphi) \notin \mathbf{s}$-NCA for $\mathbf{s} \in \{\mathrm{AD}_{\neq\emptyset}, \mathrm{CP}, \mathrm{ST}, \mathrm{PR}\}$. ❑

Theorem 25. ST-PSA *and* ST-PEXSA *are* Σ_2^p-*hard.*

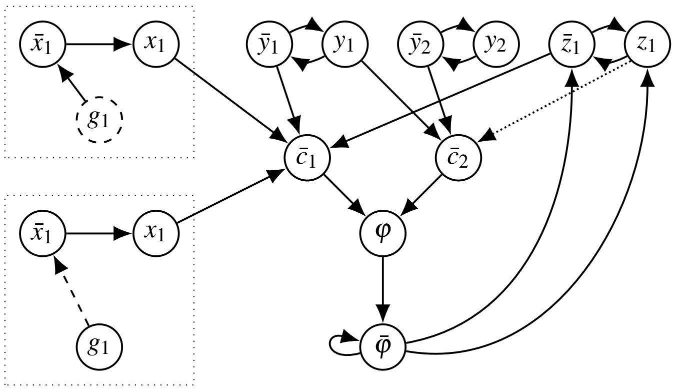


Figure 8: PR-PSA instance created from clauses $c_1 = x_1 \lor \neg y_1 \lor \neg z_1$ and $c_2 = y_1 \lor \neg y_2 \lor z_1$ using the construction of Definition 26. A slight modification that uses $c'_2 = y_1 \lor \neg y_2$ instead of c_2 can be obtained by excluding the dotted attack $(z_1, \bar{c}_2)$.

Proof. We reduce from Σ_2SAT. Let (φ, X, Y) be a Σ_2SAT instance.

If $(\varphi, X, Y) \in \Sigma_2$SAT, we have that $(\exists \tau_X)(\forall \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{false}]$, so by Lemma 18, there is a completion AF^{τ_X} of the corresponding incomplete argumentation framework *IAF* where $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\bar{\varphi}\} \cup \{c_i \mid \nexists d \in \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] : (d, c_i) \in \mathscr{R}^{\tau_X}\}$ is stable for any choice of τ_Y. There clearly can be no stable extension other than these, so $(IAF, \bar{\varphi}) \in$ ST-PSA and $(IAF, \bar{\varphi}) \in$ ST-PEXSA.

If $(\varphi, X, Y) \notin \Sigma_2$SAT, we have that $(\forall \tau_X)(\exists \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{true}]$, so for all completions of the corresponding incomplete argumentation framework *IAF*, there is some τ_Y such that the set $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is stable. Therefore, $(IAF, \bar{\varphi}) \notin$ ST-PSA. and $(IAF, \bar{\varphi}) \notin$ ST-PEXSA. ❑

To prove Σ_3^p-hardness of PR-PSA, we extend the translation of Definition 14 to incorporate the third set Z of variables that occurs in a Σ_3SAT instance (φ, X, Y, Z). This translation follows the same idea as that used by Dunne and Bench-Capon [43] to show Π_2^p-hardness of PR-SA.

Definition 26. Let (φ, X, Y, Z) be an instance of Σ_3SAT. Let $IAF' = \langle \mathscr{A}', \mathscr{A}'^?, \mathscr{R}', \mathscr{R}'^? \rangle$ be an incomplete argumentation framework created for φ, X, and Y according to Definition 14. We extend IAF' to IAF by adding literal arguments and corresponding attacks against clause arguments for all literals in Z, and by letting $\bar{\varphi}$ attack itself and every literal argument from Z. Formally, $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ with $\mathscr{A}^? = \mathscr{A}'^?$, $\mathscr{R}^? = \mathscr{R}'^?$,

$$\begin{aligned} \mathscr{A} =& \mathscr{A}' \cup \{z_i, \bar{z}_i \mid z_i \in Z\}, \text{ and} \\ \mathscr{R} =& \mathscr{R}' \cup \{(z_i, \bar{z}_i), (\bar{z}_i, z_i) \mid z_i \in Z\} \\ & \cup \{(z_k, \bar{c}_i) \mid z_k \text{ in } c_i\} \cup \{(\bar{z}_k, \bar{c}_i) \mid \neg z_k \text{ in } c_i\} \\ & \cup \{(\bar{\varphi}, \bar{\varphi})\} \cup \{(\bar{\varphi}, z_i), (\bar{\varphi}, \bar{z}_i) \mid z_i \in Z\}. \end{aligned}$$

Example 27. We extend the QSAT instance from Example 15 to a Σ_3SAT instance $(\varphi, \{x_1\}, \{y_1, y_2\}, \{z_1\})$, where $\varphi = c_1 \land c_2$ with $c_1 = x_1 \lor \neg y_1 \lor \neg z_1$ and $c_2 = y_1 \lor \neg y_2 \lor z_1$. Figure 8 displays a graph representation of the incomplete argumentation framework created for this instance of Σ_3SAT: For $\tau_X(x_1) = \texttt{true}$, any assignment τ_Y on $\{y_1, y_2\}$, and $\tau_Z(z_1) = \texttt{true}$, we have $\varphi[\tau_X, \tau_Y, \tau_Z] = \texttt{true}$. Accordingly, in the completion AF^{τ_X} all preferred extensions are of the form $\{g_1, x_1, \varphi\} \cup \mathscr{A}[\tau_Y, \tau_Z]$ for some τ_Y and a corresponding τ_Z, so φ is skeptically preferred.

When changing c_2 to $c'_2 = y_1 \lor \neg y_2$ and $\varphi' = c_1 \land c'_2$, we obtain a "no" instance. For τ_Y with $\tau_Y(y_1) = \texttt{false}$ and $\tau_Y(y_2) = \texttt{true}$, along with any assignments τ_X and τ_Z, we have $\varphi'[\tau_X, \tau_Y, \tau_Z] = \texttt{false}$. In the corresponding argumentation framework, in both completions either $\{g_1, x_1, \bar{y}_1, y_2, \bar{c}'_2\}$ or $\{g_1, \bar{x}_1, \bar{y}_1, y_2, \bar{c}'_2\}$ is a preferred extension that does not include φ, so φ is not skeptically preferred.

Lemma 28. *Given a* Σ_3SAT *instance* (φ, X, Y, Z) *and an assignment* τ_X *on* X*, let IAF be an incomplete argumentation framework created for* (φ, X, Y, Z) *following Definition 26, and let* AF^{τ_X} *be its completion corresponding to* τ_X*.* AF^{τ_X} *has a preferred extension that does not contain* φ *if and only if* $(\exists \tau_Y)(\forall \tau_Z)[\varphi[\tau_X, \tau_Y, \tau_Z] = \texttt{false}]$.

Proof. Assume that $(\exists\tau_Y)(\forall\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{false}]$. Let τ_Y be an assignment on Y satisfying $(\forall\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{false}]$ and let τ_Z be any assignment on Z. Since $\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{false}$, there is a clause c_k in φ that is not satisfied by the total assignment given by τ_X, τ_Y, and τ_Z, which means that argument φ is not defended against the respective clause argument $\bar{c}_k$ by $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y,\tau_Z]$. However, the subset $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y]$ of the literal arguments is admissible. This admissible set can only be enlarged by adding appropriate clause arguments, namely to the set $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y] \cup \{\bar{c}_i \mid \nexists d \in (\mathscr{A}^{\tau_X}[\tau_X,\tau_Y] \cup \{z_k,\bar{z}_k \mid z_k \in Z\}) : (d,\bar{c}_i) \in \mathscr{R}^{\tau_X}\}$. This set is a preferred set that does not contain φ, which proves the claim.

For the other direction, assume that $(\forall\tau_Y)(\exists\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{true}]$. Let τ_Y be any assignment on Y and let τ_Z be an assignment on Z that satisfies $\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{true}$. We prove that every preferred extension of AF^{τ_X} contains φ. For the sake of contradiction, assume that there is a preferred extension $\mathscr{E}$ of AF^{τ_X} with $\varphi \notin \mathscr{E}$. Then, there must be an argument in $\mathscr{E}$ that attacks φ, or there must be a clause argument $\bar{c}_k$ against which φ is not defended by $\mathscr{E}$. Since the only arguments that attack φ are the clause arguments, and since $\mathscr{E}$ must be conflict-free, the first condition is a special case of the second and we may assume that there is a clause argument $\bar{c}_k$ that is unattacked by $\mathscr{E}$. We know the following about $\mathscr{E}$:

- We can again apply Lemma 16 and know that $\mathscr{A}^{\tau_X}[\tau_X]$ is a subset of the grounded extension and therefore contained in every preferred extension, in particular, $\mathscr{A}^{\tau_X}[\tau_X] \subseteq \mathscr{E}$.
- It is clear that every preferred extension of AF^{τ_X} contains exactly one of the arguments y_i or $\bar{y}_i$ for each variable $y_i \in Y$. This means that there is some assignment τ_Y on Y such that $\mathscr{A}^{\tau_X}[\tau_Y] \subseteq \mathscr{E}$.
- We know that $\mathscr{E}$ cannot contain any arguments representing Z variables, since these would require $\varphi \in \mathscr{E}$ to defend them against $\bar{\varphi}$.

We can summarize that $\mathscr{E}$ must consist of $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y]$ (for some τ_Y) and at least one clause argument that is unattacked by $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y]$. However, we know by our original assumption that $(\forall\tau_Y)(\exists\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{true}]$, which means that, for the fixed given assignment τ_X on X and for *every* assignment τ_Y on Y, we can find an assignment τ_Z on Z such that $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y,\tau_Z]$ attacks all clause arguments. $\mathscr{E}$ cannot defend clause arguments against the attacks by Z arguments, so $\mathscr{E}$ cannot contain any clause arguments. But since $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y,\tau_Z]$ attacks all clause arguments, it defends φ, which makes $\mathscr{A}^{\tau_X}[\tau_X,\tau_Y,\tau_Z] \cup \{\varphi\}$ a stable extension, contradicting the assumption that $\mathscr{E}$ was preferred. ❑

We are now ready to prove Σ_3^p-hardness of PR-PSA.

Theorem 29. PR-PSA *is Σ_3^p-hard.*

Proof. We reduce Σ_3SAT to PR-PSA. Let (φ,X,Y,Z) be a Σ_3SAT instance. If $(\varphi,X,Y,Z) \in \Sigma_3$SAT, then it holds that $(\exists\tau_X)(\forall\tau_Y)(\exists\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{true}]$. Let *IAF* be the incomplete argumentation framework corresponding to (φ,X,Y,Z), let τ_X be any assignment that satisfies $(\forall\tau_Y)(\exists\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{true}]$, and let AF^{τ_X} be the completion of *IAF* representing τ_X. By Lemma 28, we know that every preferred set in AF^{τ_X} contains φ, so $(IAF,\varphi) \in$ PR-PSA.

If $(\varphi,X,Y,Z) \notin \Sigma_3$SAT, we have that $(\forall\tau_X)(\exists\tau_Y)(\forall\tau_Z)[\varphi[\tau_X,\tau_Y,\tau_Z] = \texttt{false}]$. Let *IAF* be the incomplete argumentation framework corresponding to (φ,X,Y,Z), let τ_X be any assignment on X, and let AF^{τ_X} be the completion of *IAF* representing τ_X. By Lemma 28, we know that AF^{τ_X} has a preferred extension that does not contain φ, so $(IAF,\varphi) \notin$ PR-PSA. ❑

We turn now to the problems $\text{AD}_{\neq\emptyset}$-SA, $\text{AD}_{\neq\emptyset}$-PSA, and $\text{AD}_{\neq\emptyset}$-PEXSA, for which the generic translation of Definition 14 must be adapted, too. All completions of the incomplete argumentation frameworks generated by the translation of Definition 14 have admissible sets consisting of only grounded arguments g_i and certain literal arguments, so the target arguments φ and $\bar{\varphi}$ will never be skeptically accepted with respect to the $\text{AD}_{\neq\emptyset}$ semantics. Thus, for the problems $\text{AD}_{\neq\emptyset}$-SA, $\text{AD}_{\neq\emptyset}$-PSA, and $\text{AD}_{\neq\emptyset}$-PEXSA, we use the adapted translation given in Definition 30, in which $\bar{\varphi}$ additionally attacks all literal arguments and grounded arguments. This does not impair admissible sets that contain φ, but has the effect that argument $\bar{\varphi}$ is a member of *all* nonempty admissible sets unless formula φ is satisfiable.

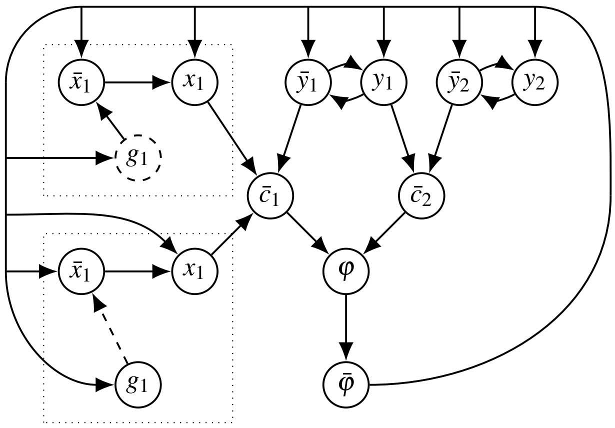


Figure 9: Visualization of either the argument-incomplete argumentation framework (not including the bottom framed part) or attack-incomplete argumentation framework (not including the top framed part) created by Definition 30 for the clauses $c_1 = x_1 \vee \neg y_1$ and $c_2 = y_1 \vee \neg y_2$ given in Example 31.

Definition 30. Let (φ, X, Y) be an instance of Σ_2SAT. Let $IAF' = \langle \mathscr{A}', \mathscr{A}'^?, \mathscr{R}', \mathscr{R}'^? \rangle$ be an incomplete argumentation framework created for it according to Definition 14. We extend IAF' to IAF by letting $\bar{\varphi}$ attack every literal argument and every grounded argument g_i. Formally, define $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ by $\mathscr{A} = \mathscr{A}'$, $\mathscr{A}^? = \mathscr{A}'^?$, $\mathscr{R}^? = \mathscr{R}'^?$, and

$$\mathscr{R} = \mathscr{R}' \cup \{(\bar{\varphi}, a) \mid a \in \{g_i, x_i, \bar{x}_i, y_j, \bar{y}_j\}\}.$$

Example 31. For formula φ in Example 15, the modified translation of Definition 30 produces the argument-incomplete or attack-incomplete argumentation framework displayed in Figure 9. The modifications leave the extension containing argument φ (Figure 7a) unaffected. Extensions that contain $\bar{\varphi}$ together with literal arguments (Figure 7b), however, are no longer possible due to the added attacks. Instead, $\{\bar{\varphi}, c_1\}$, $\{\bar{\varphi}, c_2\}$, and $\{\bar{\varphi}, c_1, c_2\}$ are (nonempty) admissible sets in *every* completion.

Lemma 32 is a variant of Lemma 18 for the adapted translation of Definition 30.

Lemma 32. *Given a* QSAT *instance* (φ, X, Y) *and an assignment* τ_X *on* X*. Let IAF be an incomplete argumentation framework created for* (φ, X, Y) *following Definition 30, and let* AF^{τ_X} *be its completion corresponding to* τ_X*.*

1. *For any assignment* τ_Y *on* Y*, let* $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y]$ *be the set of literal arguments and grounded arguments corresponding to the total assignment. If* $\varphi[\tau_X, \tau_Y] = \texttt{true}$*, then* $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ *is admissible in* AF^{τ_X}*.*

2. *Let* C *be a set of arguments containing* $\bar{\varphi}$ *and any number of—but at least one—clause argument* $\bar{c}_i$*.* C *is admissible in* AF^{τ_X}*.*

3. *If* $\varphi[\tau_X, \tau_Y] = \texttt{false}$ *for all assignments* τ_Y *on* Y*, then the sets* C *as defined in the previous Part 2 are the only nonempty admissible sets in* AF^{τ_X}*.*

Proof. For Part 1, the proof of Lemma 18 applies here as well. The only difference here are the additional attacks by argument $\bar{\varphi}$, which are defended by φ.

Now assume that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ (Part 2). Let C be a set containing $\bar{\varphi}$ and any positive number of clause arguments $\bar{c}_i$. C is clearly conflict-free in all completions of IAF. Also, C has (conditionally) definite attacks against all arguments in $\left(\mathscr{A} \cup \mathscr{A}^?\right) \setminus C$: $\bar{\varphi}$ has (conditionally) definite attacks against all grounded arguments g_i and against all literal arguments, and each clause argument $\bar{c}_i$ has a definite attack against φ. Thus C is necessarily stable in IAF and, in particular, admissible in AF^{τ_X}.

Finally, assume that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for the fixed assignment τ_X on X and for all assignments τ_Y on Y (Part 3). We show that the sets C are the only nonempty admissible sets in AF^{τ_X}. For the sake of contradiction, assume that there is a nonempty admissible set A in AF^{τ_X} that is not one of the sets C.

- If $\bar{\varphi} \in A$, since A is assumed to be conflict-free, then the only possibility is $A = \{\bar{\varphi}\}$. However, $\{\bar{\varphi}\}$ cannot defend itself against φ and is thus not admissible.
- If $\bar{\varphi} \notin A$, but $\bar{c}_i \in A$ for some clause argument $\bar{c}_i$, then the literal arguments corresponding to the negations of the literals in clause c_i must be in A, too, to defend $\bar{c}_i$ against its attackers. These, in turn, must be defended against the attack by $\bar{\varphi}$, which can only be achieved by having $\varphi \in A$. However, since $\bar{c}_i$ attacks φ and A is conflict-free, this is a contradiction. So, we cannot have $\bar{c}_i \in A$ when $\bar{\varphi} \notin A$.
- If $\varphi \in A$, then A must contain the literal arguments that defend φ against all clause arguments. However, such a set of literal arguments would correspond to a satisfying assignment for the formula φ, which cannot exist for the fixed assignment τ_X, due to our assumption that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for all assignments τ_Y on Y.
- If $\varphi \notin A$, the only remaining possibility is that A consists of only literal arguments and/or grounded arguments g_i. But this set cannot defend itself against $\bar{\varphi}$ and is thus not admissible.

This concludes the proof. ❑

Using Lemma 32, we can now show hardness of $\mathrm{AD}_{\neq\emptyset}$-SA, $\mathrm{AD}_{\neq\emptyset}$-PSA, and $\mathrm{AD}_{\neq\emptyset}$-PEXSA.

Theorem 33. $\mathrm{AD}_{\neq\emptyset}$-PSA *and* $\mathrm{AD}_{\neq\emptyset}$-PEXSA *are* Σ_2^p*-hard, and* $\mathrm{AD}_{\neq\emptyset}$-SA *is* coNP*-hard.*

Proof. We reduce Σ_2SAT to $\mathrm{AD}_{\neq\emptyset}$-PSA. Let (φ, X, Y) be a Σ_2SAT instance. If $(\varphi, X, Y) \in \Sigma_2$SAT, we have that $(\exists \tau_X)(\forall \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{false}]$, so by Lemma 32, there is a completion AF^{τ_X} of the corresponding incomplete argumentation framework *IAF* where every nonempty admissible set contains $\bar{\varphi}$, so $(IAF, \bar{\varphi}) \in \mathrm{AD}_{\neq\emptyset}$-PSA and $(IAF, \bar{\varphi}) \in \mathrm{AD}_{\neq\emptyset}$-PEXSA. If $(\varphi, X, Y) \notin \Sigma_2$SAT, we have that $(\forall \tau_X)(\exists \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{true}]$, so for all completions of the corresponding incomplete argumentation framework *IAF*, there is some τ_Y such that the set $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is admissible, and thus, not every nonempty admissible set in AF^{τ_X} contains $\bar{\varphi}$. Therefore, $(IAF, \bar{\varphi}) \notin \mathrm{AD}_{\neq\emptyset}$-PSA and $(IAF, \bar{\varphi}) \notin \mathrm{AD}_{\neq\emptyset}$-PEXSA.

If we fix $X = \emptyset$, then the Σ_2SAT instance $(\varphi, \emptyset, Y)$ corresponds to a 3-UNSAT instance (φ, Y). As a special case of the previous proof, we obtain a reduction from 3-UNSAT to $\mathrm{AD}_{\neq\emptyset}$-SA, which directly provides coNP-hardness of $\mathrm{AD}_{\neq\emptyset}$-SA. ❑

Next, we adapt the construction of Definition 30 to show Π_2^p-hardness of $\mathrm{AD}_{\neq\emptyset}$-NEXSA and ST-NEXSA.

Definition 34. Let (φ, X, Y) be an instance of Π_2SAT. Let $IAF' = \langle \mathscr{A}', \mathscr{A}'^?, \mathscr{R}', \mathscr{R}'^? \rangle$ be an incomplete argumentation framework created for it according to Definition 30. We extend IAF' to IAF by letting $\bar{\varphi}$ attack itself. Formally, $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ with $\mathscr{A} = \mathscr{A}'$, $\mathscr{A}^? = \mathscr{A}'^?$, $\mathscr{R}^? = \mathscr{R}'^?$, and $\mathscr{R} = \mathscr{R}' \cup \{(\bar{\varphi}, \bar{\varphi})\}$.

Lemma 35 pinpoints the behavior of the argumentation frameworks generated by Definition 34.

Lemma 35. *Given a* QSAT *instance* (φ, X, Y) *and an assignment* τ_X *on* X*. Let IAF be an incomplete argumentation framework created for* (φ, X, Y) *following Definition 34, and let* AF^{τ_X} *be its completion corresponding to* τ_X*.*

1. *For any assignment* τ_Y *on* Y*, let* $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y]$ *be the set of literal arguments and grounded arguments corresponding to the total assignment. If* $\varphi[\tau_X, \tau_Y] = \texttt{true}$*, then* $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ *is admissible and stable in* AF^{τ_X}*.*
2. *No nonempty sets other than those of the form* $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ *for some assignment* τ_Y *can be admissible or stable in* AF^{τ_X}*. In particular, if* $\varphi[\tau_X, \tau_Y] = \texttt{false}$ *for all assignments* τ_Y *on* Y*, then* AF^{τ_X} *has no nonempty admissible set and no stable extension.*

Proof. For Part 1, the proof of Lemma 18 applies here as well and yields that $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ is admissible and stable.

Now we show that no nonempty sets other than sets of the form $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ can be admissible or stable in AF^{τ_X} (Part 2). Let $A \subseteq \mathscr{A}^{\tau_X}$ be a nonempty admissible set.

- A cannot contain the self-attacking argument $\bar{\varphi}$.

- If $\bar{c}_i \in A$ for some clause argument $\bar{c}_i$, then the literal arguments corresponding to the negations of the literals in clause c_i must be in A, too, to defend $\bar{c}_i$ against its attackers. These, in turn, must be defended against the attack by $\bar{\varphi}$, which can only be achieved by having $\varphi \in A$. However, since $\bar{c}_i$ attacks φ and A is conflict-free, this is a contradiction. So, we cannot have $\bar{c}_i \in A$.

- If $\varphi \in A$, then A must contain the literal arguments that defend φ against all clause arguments. Such a set of literal arguments corresponds to a satisfying assignment for the formula φ, so we have $A = \mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ for some τ_Y that satisfies $\varphi[\tau_X, \tau_Y] = \texttt{true}$. However, if $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for all assignments τ_Y, then such a set cannot exist for our fixed assignment τ_X on X.

- If $\varphi \notin A$, the only remaining possibility is that A consists of only literal arguments and/or grounded arguments g_i. But this set cannot defend itself against $\bar{\varphi}$ and is thus not admissible.

We showed that only sets of the form $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ can be nonempty admissible sets in AF^{τ_X}. Since every stable extension of AF^{τ_X} is also a nonempty admissible set, we also know the same for the stable semantics. This concludes the proof. ❑

Theorem 36. $\mathrm{AD}_{\neq\emptyset}$-NEXSA *and* ST-NEXSA *are* Π_2^p*-hard.*

Proof. We reduce Π_2SAT to $\mathrm{AD}_{\neq\emptyset}$-NEXSA and ST-NEXSA. Let (φ, X, Y) be a Π_2SAT instance. If $(\varphi, X, Y) \in \Pi_2$SAT, we have that $(\forall \tau_X)(\exists \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{true}]$, so by Lemma 35, for all completions of the corresponding incomplete argumentation framework *IAF*, the sets $\mathscr{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{\varphi\}$ for any τ_Y are admissible and stable, and no other nonempty sets are admissible or stable. Therefore, $(IAF, \varphi) \in \mathrm{AD}_{\neq\emptyset}$-NEXSA and $(IAF, \varphi) \in$ ST-NEXSA.

If $(\varphi, X, Y) \notin \Pi_2$SAT, we have that $(\exists \tau_X)(\forall \tau_Y)[\varphi[\tau_X, \tau_Y] = \texttt{false}]$, so by Lemma 35, there is a completion AF^{τ_X} of the corresponding incomplete argumentation framework *IAF* that has no nonempty admissible set and no stable extension, so $(IAF, \varphi) \notin \mathrm{AD}_{\neq\emptyset}$-NEXSA and $(IAF, \varphi) \notin$ ST-NEXSA. ❑

We have covered the complexity of all possible and necessary generalizations of acceptance problems in incomplete argumentation frameworks. What is left to classify is the new problem $\mathrm{AD}_{\neq\emptyset}$-EXSA for standard argumentation frameworks, which, just like its sibling ST-EXSA, is hard for the class DP of the boolean hierarchy, which lies between the first and second level of the polynomial hierarchy. To prove DP-hardness of $\mathrm{AD}_{\neq\emptyset}$-EXSA, we reduce from the canonical DP-hard problem 3-SAT-UNSAT, which is the intersection of 3-SAT and 3-UNSAT. An instance $(\varphi^1, X^1, \varphi^2, X^2)$ consists of a 3-CNF formula φ^1 on a set X^1 of variables and a 3-CNF formula φ^2 on a set X^2 of variables, and the question is whether $(\varphi^1, X^1) \in$ 3-SAT and $(\varphi^2, X^2) \in$ 3-UNSAT. Our reduction bears some similarities to the one used by Dunne and Wooldridge [44] in their proof that ST-EXSA is DP-hard. The increased difficulty of our reduction is due to avoiding unwanted $\mathrm{AD}_{\neq\emptyset}$ sets that do not include the target argument, which is not accomplished by (and not necessary for) the reduction by Dunne and Wooldridge.

Definition 37. Let $(\varphi^1, X^1, \varphi^2, X^2)$ be an instance of 3-SAT-UNSAT, and let—for $k \in \{1,2\}$—$\varphi^k = \bigwedge_i c_i^k$ and $c_i^k = \bigvee_j \alpha_j^k$ for each clause c_i^k in φ^k, where the α_j^k are the literals over X^k that occur in clause c_i^k. We define an argumentation framework representation $\langle \mathscr{A}, \mathscr{R} \rangle$ of $(\varphi^1, X^1, \varphi^2, X^2)$ as follows:

$$\begin{aligned} \mathscr{A} = &\{x_i^1, \bar{x}_i^1 \mid x_i^1 \in X^1\} \cup \{x_i^2, \bar{x}_i^2 \mid x_i^2 \in X^2\} \\ &\cup \{\bar{c}_i^1 \mid c_i^1 \text{ in } \varphi^1\} \cup \{\bar{c}_i^2 \mid c_i^2 \text{ in } \varphi^2\} \\ &\cup \{\varphi^1, \bar{\varphi}^1, \varphi^2, \bar{\varphi}^2, \psi\}; \end{aligned}$$

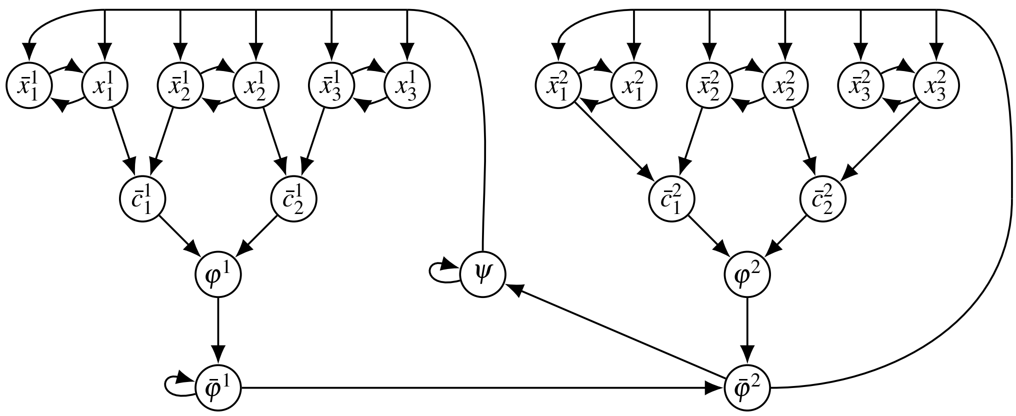


Figure 10: Visualization of the argumentation framework created by Definition 37 for the 3-SAT-UNSAT instance given in Example 38.

$$
\begin{aligned}
\mathscr{R} = &\{(x_i^1,\bar{x}_i^1),(\bar{x}_i^1,x_i^1) \mid x_i^1 \in X^1\} \cup \{(x_i^2,\bar{x}_i^2),(\bar{x}_i^2,x_i^2) \mid x_i^2 \in X^2\} \\
&\cup \{(x_k^1,\bar{c}_i^1) \mid x_k^1 \text{ in } c_i^1\} \cup \{(\bar{x}_k^1,\bar{c}_i^1) \mid \neg x_k^1 \text{ in } c_i^1\} \\
&\cup \{(x_k^2,\bar{c}_i^2) \mid x_k^2 \text{ in } c_i^2\} \cup \{(\bar{x}_k^2,\bar{c}_i^2) \mid \neg x_k^2 \text{ in } c_i^2\} \\
&\cup \{(\bar{c}_i^1,\varphi^1) \mid c_i^1 \in \varphi^1\} \cup \{(\bar{c}_i^2,\varphi^2) \mid c_i^2 \in \varphi^2\} \\
&\cup \{(\psi,x_i^1),(\psi,\bar{x}_i^1) \mid x_i^1 \in X^1\} \cup \{(\bar{\varphi}^2,x_i^2),(\bar{\varphi}^2,\bar{x}_i^2) \mid x_i^2 \in X^2\} \\
&\cup \{(\varphi^1,\bar{\varphi}^1),(\varphi^2,\bar{\varphi}^2),(\varphi^1,\varphi^1),(\varphi^1,\varphi^2),(\varphi^2,\psi),(\psi,\psi)\}.
\end{aligned}
$$

Again, we identify an assignment τ_S on a set $S = \{s_1,\dots,s_{|S|}\} \subseteq (X^1 \cup X^2)$ of variables with a set $\mathscr{A}[\tau_S] = \{s_i \mid \tau_S(s_i) = \texttt{true}\} \cup \{\bar{s}_i \mid \tau_S(s_i) = \texttt{false}\}$ of arguments.

Example 38. Let $(\varphi^1, X^1, \varphi^2, X^2)$ be a 3-SAT-UNSAT instance with $X^1 = \{x_1^1, x_2^1, x_3^1\}$, $X^2 = \{x_1^2, x_2^2, x_3^2\}$, $\varphi^1 = c_1^1 \wedge c_2^1$ with $c_1^1 = x_1^1 \vee \neg x_2^1$ and $c_2^1 = x_2^1 \vee \neg x_3^1$, and $\varphi^2 = c_1^2 \wedge c_2^2$ with $c_1^2 = \neg x_1^2 \vee \neg x_2^2$ and $c_2^2 = x_2^2 \vee x_3^2$. The argumentation framework created for $(\varphi^1, X^1, \varphi^2, X^2)$ by the translation of Definition 37 is displayed in Figure 10.

Lemma 39. *Let $(\varphi^1, X^1, \varphi^2, X^2)$ be an instance of* 3-SAT-UNSAT*, and let $\langle \mathscr{A}, \mathscr{R} \rangle$ be the argumentation framework created for it by Definition 37.*

1. $(\exists \tau_{X^1})[\varphi^1[\tau_{X^1}] = \texttt{true}]$ *if and only if there is a nonempty admissible set $\mathscr{E}$ in $\langle \mathscr{A}, \mathscr{R} \rangle$ with $\bar{\varphi}^2 \in \mathscr{E}$ (i.e., $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \in \mathrm{AD}_{\neq\emptyset}$-CA).*

2. $(\exists \tau_{X^2})[\varphi^2[\tau_{X^2}]] = \texttt{true}$ *if and only if there is a nonempty admissible set $\mathscr{E}$ in $\langle \mathscr{A}, \mathscr{R} \rangle$ with $\bar{\varphi}^2 \notin \mathscr{E}$ (i.e., $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \notin \mathrm{AD}_{\neq\emptyset}$-SA).*

Proof. Assume that $(\exists \tau_{X^1})[\varphi^1[\tau_{X^1}] = \texttt{true}]$ (Part 1). Let C^2 be a set of arguments containing $\bar{\varphi}^2$ and any number of—but at least one—clause argument $\bar{c}_i^2$ representing a clause in φ^2. Then the set $\mathscr{A}[\tau_{X^1}] \cup \{\varphi^1\} \cup C^2$ is admissible, since there are no attacks among these arguments, and because arguments $\mathscr{A}[\tau_{X^1}] \cup \{\varphi^1\}$ defeat $\bar{\varphi}^1$ and all arguments $\bar{c}_j^1$, and because arguments C^2 defeat ψ, φ^2, and all arguments x_k^1, $\bar{x}_k^1$ (analogously to Lemma 18). If instead $(\forall \tau_{X^1})[\varphi^1[\tau_{X^1}] = \texttt{false}]$, then no admissible set can defend φ^1 against all clause arguments $\bar{c}_i^1$. Since φ^1 is the only argument that can defend $\bar{\varphi}^2$ against $\bar{\varphi}^1$, however, this means that $\bar{\varphi}^2$ cannot be contained in any admissible set.

Now assume that $(\exists \tau_{X^2})[\varphi^2[\tau_{X^2}] = \texttt{true}]$ (Part 2). Then the set $\mathscr{A}[\tau_{X^2}] \cup \{\varphi^2\}$ is admissible (again, analogously to Lemma 18), so not every nonempty admissible set in $\langle \mathscr{A}, \mathscr{R} \rangle$ contains the target argument $\bar{\varphi}^2$. If instead $(\forall \tau_{X^2})[\varphi^2[\tau_{X^2}] = \texttt{false}]$, we prove that every admissible set must include the target argument $\bar{\varphi}^2$.

- Firstly, the self-attacking arguments $\bar{\varphi}^1$ and ψ can clearly never be members of admissible sets.

- The literal arguments $\{x_i^2, \bar{x}_i^2 \mid x_i^2 \in X^2\}$ need to be defended against the attack by $\bar{\varphi}^2$ in order to appear in admissible sets—however, the only argument that could provide this defending attack is φ^2, and we know that no admissible set can defend φ^2 against all clause arguments $\bar{c}_i^2$, since τ_{X^2} is unsatisfiable. Therefore, neither φ^2 nor any literal argument in $\{x_i^2, \bar{x}_i^2 \mid x_i^2 \in X^2\}$ can be in an admissible set.

- A set of arguments containing a clause argument $\bar{c}_i^2$ must also defend $\bar{c}_i^2$ against the attacks by literal arguments. The only candidates for this are other literal arguments in $\{x_i^2, \bar{x}_i^2 \mid x_i^2 \in X^2\}$ or $\bar{\varphi}^2$, and since we ruled out literal arguments in the previous step, the only option is that the admissible set must contain $\bar{\varphi}^2$.

- Literal arguments in $\{x_i^1, \bar{x}_i^1 \mid x_i^1 \in X^1\}$ must be defended against ψ in order to be members of admissible sets, which can only be done by $\bar{\varphi}^2$.

- Again, a set of arguments containing a clause argument $\bar{c}_i^1$ must defend $\bar{c}_i^1$ against the attacks by literal arguments. The only candidates for this are other literal arguments in $\{x_i^1, \bar{x}_i^1 \mid x_i^1 \in X^1\}$ or ψ. For the latter, we know that ψ can never occur in admissible sets, and for the former, we determined that this also requires $\bar{\varphi}^2$ to be in the admissible set.

- A set of arguments containing argument φ^1 must defend it against all clause arguments $\bar{c}_i^1$, which can only be done by literal arguments in $\{x_i^1, \bar{x}_i^1 \mid x_i^1 \in X^1\}$, which in turn require $\bar{\varphi}^2$ to be in the admissible set, too.

This completes the proof. ❑

Theorem 40. $\text{AD}_{\neq\emptyset}$-ExSA *is* DP-*hard.*

Proof. We reduce from 3-SAT-UNSAT. Let $(\varphi^1, X^1, \varphi^2, X^2)$ be a 3-SAT-UNSAT instance.

If $(\varphi^1, X^1, \varphi^2, X^2) \in$ 3-SAT-UNSAT, we have that $(\exists \tau_{X^1})[\varphi[\tau_{X^1}] = \texttt{true}]$ and $(\forall \tau_{X^2})[\varphi[\tau_{X^2}] = \texttt{false}]$, so by Lemma 39, there is a nonempty admissible set $\mathscr{E}$ in $\langle \mathscr{A}, \mathscr{R} \rangle$ with $\bar{\varphi}^2 \in \mathscr{E}$ (Part 1) and there is no nonempty admissible set $\mathscr{E}$ in $\langle \mathscr{A}, \mathscr{R} \rangle$ with $\bar{\varphi}^2 \notin \mathscr{E}$ (negation of Part 2), which means $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \in \text{AD}_{\neq\emptyset}$-CA and $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \in \text{AD}_{\neq\emptyset}$-SA, so $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \in \text{AD}_{\neq\emptyset}$-ExSA.

If $(\varphi^1, X^1, \varphi^2, X^2) \notin$ 3-SAT-UNSAT, we have that $(\forall \tau_{X^1})[\varphi[\tau_{X^1}] = \texttt{false}]$ or $(\exists \tau_{X^2})[\varphi[\tau_{X^2}] = \texttt{true}]$, so by Lemma 39, it holds that $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \notin \text{AD}_{\neq\emptyset}$-CA or $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \notin \text{AD}_{\neq\emptyset}$-SA, so $(\langle \mathscr{A}, \mathscr{R} \rangle, \bar{\varphi}^2) \notin \text{AD}_{\neq\emptyset}$-ExSA. ❑

4. Preservation of Extensions

In this section, we describe atomic changes to completions (adding or removing a single uncertain argument or attack) that are guaranteed to be redundant from the point of view of preserving a given **s** extension of the completion. As we will show later in Section 6–7, these observations will prove to be crucial in designing efficient algorithms for the acceptance problems with complexity beyond NP (assuming, as is widely believed, that the polynomial hierarchy does not collapse to NP). Related to the observations presented in this section, there is a substantial amount of research on the dynamic aspects of AFs (see [39] for an overview), in particular on the impact of the addition and removal of arguments and attacks on the semantics, which we overview in Section 8. The preservation of stable and preferred extensions when removing and adding attacks was first studied by Rienstra et al. [77] (in terms of labelings); our results for these semantics naturally coincide, and we further extend the results to (nonempty) admissible sets.

For an incomplete argumentation framework *IAF*, a completion $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ of *IAF*, a semantics **s**, and an extension $\mathscr{E} \in \mathbf{s}(AF^*)$, we denote $\text{IN}(\mathscr{E}) = \mathscr{E}$, $\text{OUT}(\mathscr{E}) = \{a \in \mathscr{A}^* \mid (b,a) \in \mathscr{R}^* \text{ with } b \in \text{IN}(\mathscr{E})\}$, and $\text{UNDEC}(\mathscr{E}) = \mathscr{A}^* \setminus (\text{IN}(\mathscr{E}) \cup \text{OUT}(\mathscr{E}))$.

We begin by considering adding an argument $a \in \mathscr{A}^? \setminus \mathscr{A}^*$ to the completion. If there is a definite attack $(b,a) \in \mathscr{R}$ with $b \in \text{IN}(\mathscr{E})$, then any attacks by a against arguments in $\mathscr{E}$ would be defended by $\mathscr{E}$ in the modified completion, which ensures that $\mathscr{E}$ stays an extension, both under nonempty admissible and stable semantics.

Proposition 41. *Let* $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ *be an incomplete AF,* $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ *a completion of IAF,* $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, $\mathscr{E} \in \mathbf{s}(AF^*)$*, and* $a \in \mathscr{A}^? \setminus \mathscr{A}^*$*. If there exists* $(b,a) \in \mathscr{R}$ *with* $b \in \text{IN}(\mathscr{E})$*, then* $\mathscr{E} \in \mathbf{s}(AF')$ *for* $AF' = \langle \mathscr{A}^* \cup \{a\}, \mathscr{R}^* \cup \mathscr{R}|_{\mathscr{A}^* \cup \{a\}} \rangle$.

Proof. Clearly, the set $\mathcal{E} \subseteq \mathcal{A}^*$ is still conflict-free in AF'.

- If $\mathbf{s} = \text{AD}_{\neq\emptyset}$, we know that every attack $(c,d) \in \mathcal{R}^*$ against $\mathcal{E}$ with $c \in \mathcal{A}^*$ and $d \in \mathcal{E}$ is defended by $\mathcal{E}$ since $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, and the only new attacks against $\mathcal{E}$ that might have been added in AF' are attacks by a, which are defended by $b \in \mathcal{E}$ by assumption. Therefore, $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF')$.
- If $\mathbf{s} = \text{ST}$, since $\mathcal{E} \in \text{ST}(AF^*)$, for each $c \in \mathcal{A}^* \setminus \mathcal{E}$ there exists an attack $(d,c) \in \mathcal{R}^*$ with $d \in \mathcal{E}$. Also, by assumption $(b,a) \in \mathcal{R}$ and $b \in \mathcal{E}$, and thus $\mathcal{E} \in \text{ST}(AF')$.

This completes the proof. ❑

We continue by considering removing an argument $a \in \mathcal{A}^? \cap \mathcal{A}^*$ from the completion. In particular, removing arguments (along with incident attacks) that are members of $\text{OUT}(\mathcal{E})$ or of $\text{UNDEC}(\mathcal{E})$ has no effect on the current extension $\mathcal{E}$.

Proposition 42. *Let $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ a completion of IAF, $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, $\mathcal{E} \in \mathbf{s}(AF^*)$, and $a \in \mathcal{A}^? \cap \mathcal{A}^*$. If $a \notin \text{IN}(\mathcal{E})$, then $\mathcal{E} \in \mathbf{s}(AF')$ for $AF' = \langle \mathcal{A}^* \setminus \{a\}, \mathcal{R}^*|_{\mathcal{A}^* \setminus \{a\}} \rangle$.*

Proof. Again, the set $\mathcal{E} \subseteq \mathcal{A}^*$ is still conflict-free in AF'.

- If $\mathbf{s} = \text{AD}_{\neq\emptyset}$, since $a \notin \mathcal{E}$ and $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, $\mathcal{E}$ still defends all its members in AF' and thus $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF')$.
- If $\mathbf{s} = \text{ST}$, since $\mathcal{E} \in \text{ST}(AF^*)$, for each $c \in \mathcal{A}^* \setminus \mathcal{E}$ there exists an attack $(d,c) \in \mathcal{R}^*$ with $d \in \mathcal{E}$. Since $a \notin \mathcal{E}$, these attacks still exist in AF'. Thus $\mathcal{E} \in \text{ST}(AF')$.

This completes the proof. ❑

We continue with an illustrative example on an application of Proposition 42.

Example 43. Consider the incomplete argumentation framework $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ from Example 8 illustrated in Figure 3. Further, consider the completion in Figure 4a, including both uncertain arguments a and g and the uncertain attack (f,d). By Example 10, $\mathcal{E} = \{a,c,f\}$ is a nonempty admissible extension of this completion, and we have $\text{IN}(\mathcal{E}) = \{a,c,f\}$, $\text{OUT}(\mathcal{E}) = \{b,d,e\}$, and $\text{UNDEC}(\mathcal{E}) = \{g\}$. Removing the argument g results in the completion illustrated in Figure 4c. By Proposition 42, since $g \notin \text{IN}(\mathcal{E})$, we know that $\mathcal{E}$ is an extension of this completion.

Next, we identify redundant attacks. Removing an uncertain attack $(b,a) \in \mathcal{R}^? \cap \mathcal{R}^*$ with the source $b \in \text{OUT}(\mathcal{E})$ has no effect on the extension $\mathcal{E}$, and neither does removing an attack with the target $a \in \text{IN}(\mathcal{E})$. Further, the fact that $\mathcal{E}$ is admissible is not changed by removing attacks between $\text{UNDEC}(\mathcal{E})$ arguments or from $\text{UNDEC}(\mathcal{E})$ to $\text{OUT}(\mathcal{E})$ arguments (when $\mathcal{E}$ is stable, $\text{UNDEC}(\mathcal{E}) = \emptyset$).

Proposition 44. *Let $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ a completion of IAF, $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, $\mathcal{E} \in \mathbf{s}(AF^*)$, and $(b,a) \in \mathcal{R}^? \cap \mathcal{R}^*$. If $b \notin \text{IN}(\mathcal{E})$ or $a \notin \text{OUT}(\mathcal{E})$, then $\mathcal{E} \in \mathbf{s}(AF')$ for $AF' = \langle \mathcal{A}^*, \mathcal{R}^* \setminus \{(b,a)\} \rangle$.*

Proof. Clearly, the set $\mathcal{E} \subseteq \mathcal{A}^*$ is still conflict-free in AF'. Suppose $b \notin \text{IN}(\mathcal{E})$, that is, $b \notin \mathcal{E}$.

- $\mathbf{s} = \text{AD}_{\neq\emptyset}$: Since $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, each attack $(a',c) \in \mathcal{R}^*$ where $c \in \mathcal{E}$ and $a' \in \mathcal{A}^*$ is countered by an attack $(b',a') \in \mathcal{R}^*$ with $b' \in \mathcal{E}$. This holds in AF', since by assumption $b \notin \mathcal{E}$, and hence the counterattack is not removed. Thus $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF')$.
- $\mathbf{s} = \text{ST}$: Since $\mathcal{E} \in \text{ST}(AF^*)$, for each $a' \in \mathcal{A}^* \setminus \mathcal{E}$ there is an attack $(b',a') \in \mathcal{R}^*$ with $b' \in \mathcal{E}$. This holds in AF', since by assumption $b \notin \mathcal{E}$. Thus $\mathcal{E} \in \text{ST}(AF')$.

Suppose $a \notin \text{OUT}(\mathcal{E})$, that is, there is no attack $(e,a) \in \mathcal{R}^*$ with $e \in \mathcal{E}$.

- $\mathbf{s} = \text{AD}_{\neq\emptyset}$: Since $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, each attack $(a',c) \in \mathcal{R}^*$ where $c \in \mathcal{E}$ and $a' \in \mathcal{A}^*$ is countered by an attack $(b',a') \in \mathcal{R}^*$ with $b' \in \mathcal{E}$. If for some attack we would have $a' = a$, this would be a contradiction with the assumption that there is no attack $(e,a) \in \mathcal{R}^*$ with $e \in \mathcal{E}$. That is, the counterattacks are not removed, and hence, $\mathcal{E} \in \text{AD}_{\neq\emptyset}(AF')$.

- $\mathbf{s} = \text{ST}$: Since $\mathscr{E} \in \text{ST}(AF^*)$, for each $a' \in \mathscr{A}^* \setminus \mathscr{E}$ there is an attack $(b',a') \in \mathscr{R}^*$ with $b' \in \mathscr{E}$. Again, if we would have $a' = a$, this would be a contradiction with the assumption that there is no attack $(e,a) \in \mathscr{R}^*$ with $e \in \mathscr{E}$. Thus $\mathscr{E} \in \text{ST}(AF')$.

This completes the proof. ❑

Finally, we consider adding uncertain attacks to the completion. Adding an uncertain attack $(b,a) \in \mathscr{R}^? \setminus \mathscr{R}^*$ is only relevant if the source $b \notin \text{OUT}(\mathscr{E})$ and if the target $a \in \text{IN}(\mathscr{E})$, assuming $b,a \in \mathscr{A}^*$. This is due to the fact that attacks by $\text{OUT}(\mathscr{E})$ arguments are defended and thus inactivated by $\mathscr{E}$, and that adding any further attacks against $\text{OUT}(\mathscr{E})$ arguments has no effect on $\mathscr{E}$. The admissibility of $\mathscr{E}$ is not changed by adding attacks between $\text{UNDEC}(\mathscr{E})$ arguments or from $\text{IN}(\mathscr{E})$ arguments to $\text{UNDEC}(\mathscr{E})$ arguments. Note that we assume $b,a \in \mathscr{A}^*$ here, since if $b \notin \mathscr{A}^*$ or $a \notin \mathscr{A}^*$, we would first need to add the argument in order to add the attack, and we only consider atomic changes.

Proposition 45. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ a completion of IAF, $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, $\mathscr{E} \in \mathbf{s}(AF^*)$, and $(b,a) \in \mathscr{R}^? \setminus \mathscr{R}^*$ with $b,a \in \mathscr{A}^*$. If $b \in \text{OUT}(\mathscr{E})$ or $a \notin \text{IN}(\mathscr{E})$, then $\mathscr{E} \in \mathbf{s}(AF')$ for $AF' = \langle \mathscr{A}^*, \mathscr{R}^* \cup \{(b,a)\} \rangle$.*

Proof. The set $\mathscr{E} \subseteq \mathscr{A}^*$ is still conflict-free in AF'. Suppose $b \in \text{OUT}(\mathscr{E})$, that is, there exists an attack $(e,b) \in \mathscr{R}^*$ with $e \in \mathscr{E}$.

- $\mathbf{s} = \text{AD}_{\neq\emptyset}$: Since $\mathscr{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, each attack $(a',c) \in \mathscr{R}^*$ where $c \in \mathscr{E}$ and $a' \in \mathscr{A}^*$ is countered by an attack $(b',a') \in \mathscr{R}^*$ with $b' \in \mathscr{E}$. If $a \in \mathscr{E}$, by assumption there exists an attack $(e,b) \in \mathscr{R}^*$ with $e \in \mathscr{E}$, so the attack (b,a) is countered as well. Hence, $\mathscr{E} \in \text{AD}_{\neq\emptyset}(AF')$.

- $\mathbf{s} = \text{ST}$: Since $\mathscr{E} \in \text{ST}(AF^*)$, for each $a' \in \mathscr{A}^* \setminus \mathscr{E}$ there is an attack $(b',a') \in \mathscr{R}^*$ with $b' \in \mathscr{E}$. This holds in AF', since by assumption $b \notin \mathscr{E}$. Thus $\mathscr{E} \in \text{ST}(AF')$.

Suppose $a \notin \text{IN}(\mathscr{E})$, that is, $a \notin \mathscr{E}$.

- $\mathbf{s} = \text{AD}_{\neq\emptyset}$: Since $\mathscr{E} \in \text{AD}_{\neq\emptyset}(AF^*)$, each attack $(a',c) \in \mathscr{R}^*$ where $c \in \mathscr{E}$ and $a' \in \mathscr{A}^*$ is countered by an attack $(b',a') \in \mathscr{R}^*$ with $b' \in \mathscr{E}$. Since by assumption $a \notin \mathscr{E}$, adding the attack (b,a) has no effect on the admissibility of $\mathscr{E}$. Hence, $\mathscr{E} \in \text{AD}_{\neq\emptyset}(AF')$.

- $\mathbf{s} = \text{ST}$: Since $\mathscr{E} \in \text{ST}(AF^*)$, for each $a' \in \mathscr{A}^* \setminus \mathscr{E}$ there is an attack $(b',a') \in \mathscr{R}^*$ with $b' \in \mathscr{E}$. This holds in AF', and hence $\mathscr{E} \in \text{ST}(AF')$.

This completes the proof. ❑

We continue with an illustrative example on an application of Proposition 45.

Example 46. Consider the incomplete argumentation framework $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ from Example 8, illustrated in Figure 3. Further, consider the completion in Figure 4h, excluding both uncertain arguments a and g and the uncertain attack (f,d). By Example 10, $\mathscr{E} = \{b,e,f\}$ is a stable extension of this completion and we have $\text{IN}(\mathscr{E}) = \{b,e,f\}$ and $\text{OUT}(\mathscr{E}) = \{c,d\}$. Including the attack (f,d) results in the completion illustrated in Figure 4g. By Proposition 45, since the target argument $d \notin \text{IN}(\mathscr{E})$, we know that $\mathscr{E}$ is still an extension of this completion.

In addition to admissible and stable semantics, we make use of similar results for preferred semantics, under which the picture is slightly different as the conditions on undecided arguments are more restrictive when removing arguments and changing the attack structure. Consider first adding an argument in $\mathscr{A}^? \setminus \mathscr{A}^*$. If it is definitely attacked by an argument in the preferred extension, adding it will not affect the extension being preferred, exactly as in Proposition 41 for admissible and stable semantics.

Proposition 47. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ a completion of IAF, $\mathscr{E} \in \text{PR}(AF^*)$, and $a \in \mathscr{A}^? \setminus \mathscr{A}^*$. If there exists $(b,a) \in \mathscr{R}$ with $b \in \text{IN}(\mathscr{E})$, then $\mathscr{E} \in \text{PR}(AF')$ for $AF' = \langle \mathscr{A}^* \cup \{a\}, \mathscr{R}^* \cup \mathscr{R}|_{\mathscr{A}^* \cup \{a\}} \rangle$.*

Proof. Since now $\mathscr{E} \in \textsc{ad}(F^*)$, due to Proposition 41, $\mathscr{E} \in \textsc{ad}(AF')$, and it remains to show that there is no $\mathscr{E}' \in \textsc{ad}(AF')$ with $\mathscr{E}' \supset \mathscr{E}$. Suppose on the contrary that such a set $\mathscr{E}'$ exists. Since now $a \notin \textsc{in}(\mathscr{E}')$ as otherwise $\mathscr{E}'$ would not be conflict-free, by Proposition 42, $\mathscr{E}' \in \textsc{ad}(AF^*)$, contradicting the assumption that $\mathscr{E} \in \textsc{pr}(AF^*)$. This completes the proof. ❑

If $\mathscr{E}$ is a preferred extension, removing an argument $a \in \mathscr{A}^? \cap \mathscr{A}^*$ with $a \in \textsc{undec}(\mathscr{E})$ does not necessarily preserve $\mathscr{E}$, in contrast to Proposition 42.[1] On the other hand, the observation in Proposition 42 regarding arguments in $\textsc{out}(\mathscr{E})$ does still hold for preferred semantics.

Proposition 48. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ a completion of IAF, $\mathscr{E} \in \textsc{pr}(AF^*)$, and $a \in \mathscr{A}^? \cap \mathscr{A}^*$. If $a \in \textsc{out}(\mathscr{E})$, then $\mathscr{E} \in \textsc{pr}(AF')$ for $AF' = \langle \mathscr{A}^* \setminus \{a\}, \mathscr{R}^*|_{\mathscr{A}^* \setminus \{a\}} \rangle$.*

Proof. Since $\mathscr{E} \in \textsc{ad}(AF^*)$, via Proposition 42, $\mathscr{E} \in \textsc{ad}(AF')$. It remains to show that there is no $\mathscr{E}' \in \textsc{ad}(AF')$ with $\mathscr{E}' \supset \mathscr{E}$. Suppose on the contrary that such a set $\mathscr{E}'$ exists. Since $a \in \textsc{out}(\mathscr{E})$, there exists $b \in \mathscr{E}$ with $(b,a) \in \mathscr{R}^*$. The conditions of Proposition 41 now hold for the incomplete argumentation framework $IAF' = \langle \mathscr{A}^*, \{a\}, \mathscr{R}^*, \mathscr{R}^?|_{\mathscr{A}^* \cup \{a\}} \rangle$ and the completion AF'. Since $\mathscr{E}' \in \textsc{ad}(AF')$, this results in $\mathscr{E}' \in \textsc{ad}(AF^*)$, which contradicts our assumption $\mathscr{E} \in \textsc{pr}(AF^*)$. This completes the proof. ❑

Regarding removing and adding attacks, the results by Rienstra et al. [77] for preferred semantics are directly applicable in our context. The key difference to admissible and stable semantics is that removing and adding attacks between undecided arguments and adding attacks from the extension to undecided arguments may result in a larger admissible extension. We summarize these results in the following propositions.

Proposition 49. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ a completion of IAF, $\mathscr{E} \in \textsc{pr}(AF^*)$, and $(b,a) \in \mathscr{R}^? \cap \mathscr{R}^*$. If $b \notin \textsc{in}(\mathscr{E})$ or $a \notin \textsc{out}(\mathscr{E})$, and $b \notin \textsc{undec}(\mathscr{E})$ or $a \notin \textsc{undec}(\mathscr{E})$, then $\mathscr{E} \in \textsc{pr}(AF')$ for $AF' = \langle \mathscr{A}^*, \mathscr{R}^* \setminus \{(b,a)\} \rangle$.*

Proposition 50. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF, $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ a completion of IAF, $\mathscr{E} \in \textsc{pr}(AF^*)$, and $(b,a) \in \mathscr{R}^? \setminus \mathscr{R}^*$ with $b, a \in \mathscr{A}^*$. If $b \in \textsc{out}(\mathscr{E})$ or $a \in \textsc{out}(\mathscr{E})$, then $\mathscr{E} \in \textsc{pr}(AF')$ for $AF' = \langle \mathscr{A}^*, \mathscr{R}^* \cup \{(b,a)\} \rangle$.*

We summarize our observations about the effects of atomic changes to the set of arguments on the preservation of a given extension:

- by Propositions 41 and 47, under all three semantics $\textsc{ad}_{\neq\emptyset}$, $\textsc{st}$, and $\textsc{pr}$, adding arguments that will definitely be attacked by the current extension does not have an effect on the extension;
- by Proposition 42, under $\textsc{ad}_{\neq\emptyset}$ and $\textsc{st}$, removing arguments that are not in the current extension does not have an effect on the extension;
- by Proposition 48, under $\textsc{pr}$, this only concerns arguments that are attacked by the extension.

Regarding atomic changes to the set of attacks, Table 3 summarizes the allowed changes.

5. Encoding into SAT

We begin the algorithmic part of this work by providing encodings in boolean satisfiability (SAT) for the problem variants with first-level complexity. The encodings extend the standard encodings for argumentation semantics [20] to incomplete AFs, essentially conditioning relevant parts of the formulas on the existence of an argument or an attack. Given an incomplete argumentation framework IAF and a semantics $\mathbf{s}$, we will present propositional formulas $\varphi_{\mathbf{s}}(IAF)$ such that completions of the incomplete AF and $\mathbf{s}$ extensions of the completion are in a one-to-one correspondence with models of $\varphi_{\mathbf{s}}(IAF)$. These encodings will then form the basis for the SAT-based algorithms deciding the various

[1] In particular, removing arguments in $\textsc{undec}(\mathscr{E})$ may break odd-length cycles and thus modify the preferred extension.

Table 3: Overview of the atomic changes on the set of attacks that do not affect an extension $\mathcal{E}$ of a completion, based on Propositions 44 and 45 for $\text{AD}_{\neq\emptyset}$ and ST and on Propositions 49 and 50 for PR. Here, the symbol ✓ means that the change has no effect under all semantics $\text{AD}_{\neq\emptyset}$,ST, and PR, (✓) means that the change has no effect under semantics $\text{AD}_{\neq\emptyset}$ or ST but may have an effect under PR, and ✗ means that the change may have an effect under all semantics $\text{AD}_{\neq\emptyset}$,ST, and PR.

	adding attack (b,a)			removing attack (b,a)		
	$a \in \text{IN}(\mathcal{E})$	$a \in \text{OUT}(\mathcal{E})$	$a \in \text{UNDEC}(\mathcal{E})$	$a \in \text{IN}(\mathcal{E})$	$a \in \text{OUT}(\mathcal{E})$	$a \in \text{UNDEC}(\mathcal{E})$
$b \in \text{IN}(\mathcal{E})$	✗	✓	(✓)	✓	✗	✓
$b \in \text{OUT}(\mathcal{E})$	✓	✓	✓	✓	✓	✓
$b \in \text{UNDEC}(\mathcal{E})$	✗	✓	(✓)	✓	✓	(✓)

forms of acceptance in incomplete argumentation frameworks, including the NP- and coNP-complete problems which are solved using a single SAT solver call, as well as problems beyond NP for which we develop counterexample-guided approaches in Section 6.

Consider an input $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$. We use variables x_a and y_a for all $a \in \mathcal{A} \cup \mathcal{A}^?$ and $r_{a,b}$ for all $(a,b) \in \mathcal{R} \cup \mathcal{R}^?$, with the following interpretations:

- $y_a = \texttt{true}$ if and only if $a \in \mathcal{A}^*$,
- $r_{a,b} = \texttt{true}$ if and only if $(a,b) \in \mathcal{R}^*$, and
- $x_a = \texttt{true}$ if and only if $a \in \mathcal{E} \in \mathbf{s}(AF^*)$,

where $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ is a completion of IAF defined by the y_* and $r_{*,*}$ variables. In other words, the x_* variables encode an extension of the completion encoded by the y_* and $r_{*,*}$ variables. Now, the formula

$$\varphi_{\text{CF}}(IAF) = \bigwedge_{(a,b) \in \mathcal{R} \cup \mathcal{R}^?} \Big((y_a \wedge y_b \wedge r_{a,b}) \rightarrow (\neg x_a \vee \neg x_b) \Big)$$

encodes conflict-free sets in a completion of IAF, expressing that if two arguments and an attack between them are present in the completion, one cannot include both of the arguments in the extension. To express IAF semantics, we use additional variables z_a for each $a \in \mathcal{A} \cup \mathcal{A}^?$, binding their value to the rest of the variables via equivalences

$$\varphi_z(IAF) = \bigwedge_{a \in \mathcal{A} \cup \mathcal{A}^?} \left(z_a \leftrightarrow \bigvee_{(b,a) \in \mathcal{R} \cup \mathcal{R}^?} (x_b \wedge y_b \wedge r_{b,a}) \right).$$

Now z_a is assigned to $\texttt{true}$ if and only if a is attacked by the extension encoded by the x_* variables in the completion encoded by the y_* and $r_{*,*}$ variables. Using this and the encoding for conflict-free sets,

$$\varphi_{\text{AD}}(IAF) = \varphi_{\text{CF}}(IAF) \wedge \varphi_z(IAF) \wedge \bigwedge_{a \in \mathcal{A} \cup \mathcal{A}^?} \bigwedge_{(b,a) \in \mathcal{R} \cup \mathcal{R}^?} \Big((x_a \wedge y_a \wedge y_b \wedge r_{b,a}) \rightarrow z_b \Big)$$

encodes admissible sets by expressing that, for each argument and for each attack on the argument, if the argument is included in the extension of the completion, the attacker must be attacked by the extension. That is, the extension defends itself. Nonempty admissible semantics can now be encoded as

$$\varphi_{\text{AD}_{\neq\emptyset}}(IAF) = \varphi_{\text{AD}}(IAF) \wedge \bigvee_{a \in \mathcal{A} \cup \mathcal{A}^?} x_a.$$

The encoding for complete semantics is

$$\varphi_{\text{CP}}(IAF) = \varphi_{\text{AD}}(IAF) \wedge \bigwedge_{a \in \mathcal{A} \cup \mathcal{A}^?} \left(\bigwedge_{(b,a) \in \mathcal{R} \cup \mathcal{R}^?} \left((y_a \wedge y_b \wedge r_{b,a}) \rightarrow z_b \right) \rightarrow x_a \right),$$

stating that a complete extension is admissible, and for each argument, if the argument is defended by the extension of the completion (i.e., each attacker of the argument is attacked by the extension if the attack and the incident arguments are present in the completion), it is also included in the extension. We express stable semantics with

$$\varphi_{\text{ST}}(IAF) = \varphi_{\text{CF}}(IAF) \wedge \varphi_z(IAF) \wedge \bigwedge_{a \in \mathscr{A} \cup \mathscr{A}^?} \Big((y_a \wedge \neg x_a) \rightarrow z_a \Big),$$

which states that a stable extension is conflict-free, and for each argument, if the argument exists in the completion and is not included in the extension, it must be attacked by the extension.

Finally, for the grounded semantics we adapt the encoding due to Niskanen et al. [73], using additionally variables l_n^a for each $a \in \mathscr{A} \cup \mathscr{A}^?$ and for each integer $n \in \{1, \ldots, \lceil |\mathscr{A} \cup \mathscr{A}^?|/2 \rceil\}$, with the interpretation that $l_n^a =$ `true`if and only if $a \in F^n_{AF^*}(\emptyset)$. The upper bound on n is explained by the fact that for AFs with $|\mathscr{A}|$ arguments, the maximum number of iterations of the characteristic function to the empty set until the fixpoint is $\lceil |\mathscr{A}|/2 \rceil$. This upper bound is realized by an AF which consists of a directed path of arguments. The constraints

$$\varphi_1^a(IAF) = l_1^a \leftrightarrow \Bigg(y_a \wedge \bigwedge_{(b,a) \in \mathscr{R} \cup \mathscr{R}^?} (y_b \rightarrow \neg r_{b,a}) \Bigg),$$

$$\varphi_n^a(IAF) = l_n^a \leftrightarrow \Bigg(y_a \wedge \bigwedge_{(b,a) \in \mathscr{R} \cup \mathscr{R}^?} \Big((y_b \wedge r_{b,a}) \rightarrow \bigvee_{(c,b) \in \mathscr{R} \cup \mathscr{R}^?} (l_{n-1}^c \wedge y_c \wedge r_{c,b}) \Big) \Bigg),$$

simulate the application of the characteristic function on a completion of the input incomplete AF, and finally

$$\varphi_{\text{GR}}(IAF) = \bigwedge_{a \in \mathscr{A} \cup \mathscr{A}^?} \left(\left(x_a \leftrightarrow \left(\bigvee_{n=1}^{\lceil |\mathscr{A} \cup \mathscr{A}^?|/2 \rceil} l_n^a \right) \right) \wedge \bigwedge_{n=1}^{\lceil |\mathscr{A} \cup \mathscr{A}^?|/2 \rceil} \varphi_n^a(IAF) \right)$$

requires that an argument is included if and only if it is included in $F^n_{AF^*}(\emptyset)$ for some n, hence capturing the grounded extension of the completion AF^*.

Finally, to encode valid completions, we observe that y_a is always `true` for all definite arguments, and likewise $r_{a,b}$ is `true` for all definite attacks.[2] Further, if an uncertain argument is not included in the completion, it cannot be accepted, and all incident attacks are not included either. This information is encoded as

$$\varphi_?(IAF) = \bigwedge_{a \in \mathscr{A}} y_a \wedge \bigwedge_{(a,b) \in \mathscr{R}} r_{a,b} \wedge \bigwedge_{a \in \mathscr{A}^?} \Bigg(\neg y_a \rightarrow \Bigg(\neg x_a \wedge \bigwedge_{(a,b) \in \mathscr{R}^?} \neg r_{a,b} \wedge \bigwedge_{(b,a) \in \mathscr{R}^?} \neg r_{b,a} \Bigg) \Bigg).$$

The latter conjunction is in fact redundant, given that the $\varphi_{\mathbf{s}}(IAF)$ formulas are conditioned on the existence of an attack and incident arguments. However, as binarý clauses, these can be beneficial for a SAT solver during search.

The key properties of the encodings are summarized in the following proposition.

Proposition 51. *Let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF. Consider a query argument $a \in \mathscr{A}$, and a semantics* $\mathbf{s} \in \{\text{AD}, \text{AD}_{\neq\emptyset}, \text{CP}, \text{ST}, \text{GR}\}$.

- *The formula $\varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge x_a$ is satisfiable if and only if argument a is possibly credulously accepted under* $\mathbf{s}$.
- *The formula $\varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge (\neg x_a)$ is unsatisfiable if and only if argument a is necessarily skeptically accepted under* $\mathbf{s}$.

[2] In fact, by performing deterministic polynomial-time unit propagation (as implemented in typical SAT solvers) on the encodings with the unit clauses $\bigwedge_{a \in \mathscr{A}} y_a \wedge \bigwedge_{(a,b) \in \mathscr{R}} r_{a,b}$ expressing the definite elements, the connectives in the $\varphi_{\mathbf{s}}(IAF)$ encodings split into several subcases depending on whether arguments and attacks are definite or uncertain. Here we present them in a uniform and more compact way for representational simplicity.

6. SAT-based Algorithms

In this section we present a SAT-based approach for argument acceptance in incomplete argumentation frameworks. For problems on the first level of the polynomial hierarchy, a single call to a SAT solver suffices using the encodings presented in Section 5. However, for problems beyond NP, there presumably exists no polynomial-time reduction to SAT. Hence, we develop algorithms based on counterexample-guided abstraction refinement (CEGAR), originally proposed in the context of model checking [30, 31]. In SAT-based CEGAR, an abstraction of the problem at hand—an overapproximation of the original problem that is in NP—is used to query a SAT solver for a candidate solution to the problem. Then, a SAT solver is again used to check whether the candidate solution is an actual solution to the problem by asking it for a counterexample. If one is found, the abstraction is refined by ruling out the counterexample. Then the algorithm proceeds iteratively until the solution of the original problem is found.

For the rest of this section, let $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ be an incomplete AF and $a \in \mathscr{A}$ be the query argument.

6.1. Credulous Acceptance

We start with possible and necessary credulous acceptance. First of all, we note that due to Observation 5, possible and necessary credulous acceptance coincides for semantics $\textsc{ad}$, $\textsc{ad}_{\neq\emptyset}$, $\textsc{cp}$, and $\textsc{pr}$. Hence, to cover all semantics and both variants of credulous acceptance, it suffices to consider the problems $\mathbf{s}$-PCA and $\mathbf{s}$-NCA for $\mathbf{s} \in \{\textsc{ad}, \textsc{st}, \textsc{gr}\}$.

Possible credulous acceptance. Recall that the possible variant $\mathbf{s}$-PCA is NP-complete for all $\mathbf{s} \in \{\textsc{ad}, \textsc{st}, \textsc{gr}\}$, and due to Proposition 51, is decided by satisfying the input formula

$$\varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge x_a.$$

Note that a satisfying truth assignment also yields a completion of the input IAF and an $\mathbf{s}$ extension of the completion containing the query argument.

Necessary credulous acceptance. The necessary variant $\mathbf{s}$-NCA is Π_2^p-complete for $\mathbf{s} \in \{\textsc{ad}, \textsc{st}\}$ and coNP-complete for $\mathbf{s} = \textsc{gr}$. Due to Observations 6 and 7, $\textsc{gr}$-NCA $=$ $\textsc{gr}$-NSA $=$ $\textsc{cp}$-NSA, so necessary credulous acceptance under grounded semantics can be decided by using the procedure for necessary skeptical acceptance under complete semantics, which we cover in Section 6.2.

We are left with the problems $\textsc{ad}$-NCA and $\textsc{st}$-NCA, for which we develop a CEGAR procedure, presented as pseudocode in Algorithm 1. Based on Π_2^p-completeness, the idea is to look for a counterexample, that is, a completion

Algorithm 1 CEGAR-based necessary credulous acceptance for $\mathbf{s} \in \{\textsc{ad}, \textsc{st}\}$. Input: $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$, $a \in \mathscr{A}$.

$\varphi \leftarrow \textsc{Abstraction}_{\mathbf{s}}(IAF, \neg x_a)$
while true
    $(sat, \tau) \leftarrow \text{SAT}(\varphi)$
    if $sat =$ `true`
        $AF^* \leftarrow \textsc{Extract}(\tau)$
        $(sat, \tau') \leftarrow \text{SAT}(\textsc{Check}_{\mathbf{s}}(IAF, AF^*, x_a))$
        if $sat =$ `false` return *reject*
        $\varphi \leftarrow \varphi \wedge \textsc{Refine}(IAF, AF^*)$
    else
        if $\mathbf{s} \neq \textsc{st}$ return *accept*
        $\varphi \leftarrow (\varphi \setminus \{\neg x_a\}) \cup \{x_a\}$
        while true
            $(sat, \tau) \leftarrow \text{SAT}(\varphi)$
            if $sat =$ `false` **break**
            $AF^* \leftarrow \textsc{Extract}(\tau)$
            $\varphi \leftarrow \varphi \wedge \textsc{Refine}(IAF, AF^*)$
        if $\text{SAT}(\varphi \setminus \varphi_{\textsc{st}}(IAF)) =$ `true` return *reject*
        else return *accept*

of *IAF* which has no extension containing the query argument a. We start by solving an abstraction of the problem, given in this case by the formula (line 1)

$$\textsc{Abstraction}_{\mathbf{s}}(IAF, \neg x_a) = \varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge \neg x_a.$$

If $\textsc{Abstraction}_{\mathbf{s}}(IAF, \neg x_a)$ is satisfiable, we get a truth assignment τ from which we can extract a completion $AF^* = \textsc{Extract}(\tau)$ (lines 3 and 5) of the input IAF with $\textsc{Extract}(\tau) = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$, where

$$\mathscr{A}^* = \{a \in \mathscr{A} \cup \mathscr{A}^? \mid \tau(y_a) = \texttt{true}\}$$
$$\mathscr{R}^* = \{(a,b) \in \mathscr{R} \cup \mathscr{R}^? \mid \tau(r_{a,b}) = \texttt{true}\}$$

along with an $\mathbf{s}$ extension $\mathscr{E} \in \mathbf{s}(AF^*)$ that does not include a. We still do not know whether there is some other s extension containing a, so we check whether a is credulously accepted under $\mathbf{s}$ in AF^*, which can be done with a single SAT call on the formula (line 6)

$$\textsc{Check}_{\mathbf{s}}(IAF, AF^*, x_a) = \varphi_{\mathbf{s}}(IAF) \wedge \textsc{Completion}(IAF, AF^*) \wedge x_a,$$

where

$$\textsc{Completion}(IAF, AF^*) = \bigwedge_{a \in \mathscr{A}^*} y_a \wedge \bigwedge_{a \in (\mathscr{A} \cup \mathscr{A}^?) \setminus \mathscr{A}^*} \neg y_a \wedge \bigwedge_{(a,b) \in \mathscr{R}^*} r_{a,b} \wedge \bigwedge_{(a,b) \in (\mathscr{R} \cup \mathscr{R}^?) \setminus \mathscr{R}^*} \neg r_{a,b}$$

encodes the completion AF^* currently under consideration. Adding these unit clauses essentially reduces the encoding $\varphi_{\mathbf{s}}(IAF)$ for incomplete AFs to the respective encoding for standard AFs. If $\textsc{Check}_{\mathbf{s}}(IAF, AF^*, a)$ is unsatisfiable, there is no extension containing a in AF^*, so we successfully obtained a counterexample to necessary credulous acceptance of a, and can *reject* (line 7). If, however, the formula $\textsc{Check}_{\mathbf{s}}(IAF, AF^*, a)$ is satisfiable, our counterexample is not valid for the problem instance at hand and we need to refine the abstraction by adding the clause

$$\textsc{Refine}(IAF, AF^*) = \neg\textsc{Completion}(IAF, AF^*) = \bigvee_{a \in \mathscr{A}^*} \neg y_a \vee \bigvee_{a \in (\mathscr{A} \cup \mathscr{A}^?) \setminus \mathscr{A}^*} y_a \vee \bigvee_{(a,b) \in \mathscr{R}^*} \neg r_{a,b} \vee \bigvee_{(a,b) \in (\mathscr{R} \cup \mathscr{R}^?) \setminus \mathscr{R}^*} r_{a,b}$$

to the original SAT instance (line 8), which tells the solver that we need to find another completion of *IAF*, and iterate until we either find a counterexample to necessary credulous acceptance, or reach unsatisfiability.

If $\textsc{Abstraction}_{\mathbf{s}}(IAF, \neg x_a)$ is unsatisfiable, Proposition 51 implies that a is necessarily skeptically accepted. For $\mathbf{s} = \textsc{ad}$, this implies that a is also necessarily credulously accepted, so we *accept* (line 10). For $\mathbf{s} = \textsc{st}$, it is not necessarily the case that a is necessarily credulously accepted, since it might be the case that there are completions which have no stable extension (recall that an argument is by definition skeptically accepted if the AF has no extension). In other words, the abstraction initialized with $\textsc{Abstraction}_{\textsc{st}}(IAF, \neg x_a)$ only considers those AFs that have a stable extension not containing a.

To resolve this issue, we iteratively call a SAT solver with the input formula $(\varphi \setminus \{\neg x_a\}) \cup \{x_a\}$ consisting of $\varphi_?(IAF) \wedge \varphi_{\textsc{st}}(IAF) \wedge x_a$ and additional refinement clauses, refining the formula similarly via $\textsc{Refine}(IAF, AF^*)$ using the completion AF^* extracted from a satisfying assignment, until reaching unsatifiability (lines 11–16). Essentially, this procedure rules out all completions that possess a stable extension. Finally, we check using $\varphi \setminus \varphi_{\textsc{st}}(IAF)$ whether there still are completions; if so, we *reject* the query, since these completions have no stable extension, and otherwise we *accept* it (lines 17–18).

6.2. *Skeptical Acceptance*

We continue by presenting procedures for deciding (both variants of) possible and necessary skeptical acceptance. As explained in Section 6.1, due to Observations 6 and 7, it suffices to consider CP-NSA in order to decide necessary (credulous and skeptical) acceptance under grounded semantics. In addition, since possible (credulous and skeptical) acceptance under grounded semantics coincides with CP-PSA, CP-PSA can be solved using the procedure for GR-PCA described in Section 6.1. Finally, **s**-PSA and **s**-PEXSA coincide for semantics under which existence of an extension is guaranteed. That is, it suffices to consider **s**-PEXSA, **s**-PSA, **s**-NSA, and **s**-NEXSA for $\mathbf{s} \in \{\textsc{ad}_{\neq\emptyset}, \textsc{st}\}$, PR-PSA, CP-NSA, and PR-NSA to cover all semantics and all variants of skeptical acceptance.

Necessary skeptical acceptance. Recall that **s**-NSA is coNP-complete under $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}, \text{CP}\}$ and via Proposition 51 can be decided by checking whether the formula

$$\varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge \neg x_a \tag{1}$$

is unsatisfiable. Note that if this formula is satisfiable, we obtain a truth assignment from which we can extract a completion of the input IAF and an **s** extension of the completion not containing the query argument.

Necessary existence and skeptical acceptance. For **s**-NExSA under $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, we can use the formula (1) as an abstraction. In case of unsatisfiability, we rule all completions that have an **s** extension out of consideration, and check whether there still are completions that have no **s** extension. For details on this procedure, see the description for ST-NCA in Section 6.1. If there are no completions left, we *accept*, and otherwise we *reject*.

Possible existence and skeptical acceptance. We continue with **s**-PExSA under $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, which is Σ_2^p-complete. This hints at the idea of looking for a witness completion in which the query argument is skeptically accepted, i.e., included in every **s** extension. The CEGAR procedure is presented in pseudocode as Algorithm 2.

In this algorithm, an abstraction is now formed via the formula (line 1)

$$\text{ABSTRACTION}_{\mathbf{s}}(IAF, x_a) = \varphi_?(IAF) \wedge \varphi_{\mathbf{s}}(IAF) \wedge x_a.$$

If $\text{ABSTRACTION}_{\mathbf{s}}(IAF, x_a)$ is unsatisfiable, we can *reject* the query (line 4) since then by Proposition 51 a is not even possibly credulously accepted. If $\text{ABSTRACTION}_{\mathbf{s}}(IAF, x_a)$ is satisfiable, we extract the current completion $AF^* = \text{EXTRACT}(\tau) = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ from the satisfying truth assignment τ (line 5), where $\mathscr{A}^*$ and $\mathscr{R}^*$ are as defined on page 29. We proceed by checking whether a is skeptically accepted under **s** in AF^*. This is accomplished via a SAT call with the input formula (line 6)

$$\text{CHECK}_{\mathbf{s}}(IAF, AF^*, \neg x_a) = \varphi_{\mathbf{s}}(IAF) \wedge \text{COMPLETION}(IAF, AF^*) \wedge \neg x_a,$$

where $\text{COMPLETION}(IAF, AF^*)$ is defined on page 29.

If $\text{CHECK}_{\mathbf{s}}(IAF, AF^*, \neg x_a)$ is unsatisfiable, this proves that we have successfully found a witness completion for skeptical acceptance of a, and can *accept* (line 7). If the formula $\text{CHECK}_{\mathbf{s}}(IAF, AF^*, \neg x_a)$ is satisfiable, we have shown that there exists an extension $\mathscr{E} \in \mathbf{s}(AF^*)$ not containing the query argument, so we refine the abstraction by adding the clause $\text{REFINE}(IAF, AF^*)$ (line 8), as defined on page 29, excluding the current completion. Then, we continue iteratively until the abstraction becomes unsatisfiable or we find a witnessing completion to possible existence and skeptical acceptance.

Possible skeptical acceptance. For **s**-PSA under $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$, we can use the same procedure as for **s**-PExSA, with the following exception. If the abstraction becomes unsatisfiable, we rule out all completions that have an **s** extension, and check whether there still are completions without an **s** extension, as described in the procedure for ST-NCA in Section 6.1. If there are completions left, this implies possible skeptical acceptance, and hence we *accept*, otherwise we *reject*.

Necessary skeptical acceptance under preferred semantics. Consider now the PR-NSA problem, which is Π_2^p-complete, implying that we are again looking for a counterexample. The source of hardness is the Π_2^p-completeness

Algorithm 2 CEGAR-based possible existence and skeptical acceptance for $\mathbf{s} \in \{\text{AD}_{\neq\emptyset}, \text{ST}\}$. Input: IAF, $a \in \mathscr{A}$.

1: $\varphi \leftarrow \text{ABSTRACTION}_{\mathbf{s}}(IAF, x_a)$
2: **while** true
3: $\quad (sat, \tau) \leftarrow \text{SAT}(\varphi)$
4: $\quad$ **if** $sat =$ `false` return *reject*
5: $\quad AF^* \leftarrow \text{EXTRACT}(\tau)$
6: $\quad (sat, \tau) \leftarrow \text{SAT}(\text{CHECK}_{\mathbf{s}}(IAF, AF^*, \neg x_a))$
7: $\quad$ **if** $sat =$ `false` return *accept*
8: $\quad \varphi \leftarrow \varphi \wedge \text{REFINE}(IAF, AF^*)$

of standard skeptical acceptance under preferred semantics [43]. Motivated by this, we develop a CEGAR procedure similar to the SAT-based AF solver CEGARTIX [46]. We describe the algorithm briefly due to the similarities of the approaches. Our abstraction is based on the complete semantics via

$$\textsc{Abstraction}_{\textsc{cp}}(IAF, \neg x_a) = \varphi_?(IAF) \wedge \varphi_{\textsc{cp}}(IAF) \wedge \neg x_a.$$

If the abstraction is unsatisfiable, we *accept* the argument. A satisfying truth assignment τ yields us a completion $AF^* = \textsc{Extract}(\tau) = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ and a complete extension not containing a. Then, we iteratively subset-maximize the complete extension under the constraint that a is still not included, and finally check whether including a yields a larger complete extension. If not, we have successfully obtained a counterexample, that is, a completion where a is not skeptically accepted under preferred semantics, and we *reject*. If yes, we refine the abstraction via $\textsc{Refine}(IAF, AF^*) \vee \bigvee_{b \in \mathscr{A}^* \setminus \mathscr{E}} x_b$, where $\mathscr{E}$ is the subset-maximal complete extension we found, ruling out all subsets of it for that particular completion, and continue iteratively.

Possible skeptical acceptance under preferred semantics. The only problem complete for a class in the third level of the polynomial hierarchy is the Σ_3^p-complete PR-PSA. Reflecting the computational complexity, we are looking for a witness, that is, a completion where a is skeptically accepted under preferred semantics. For the CEGAR algorithm, we initialize the abstraction using the complete semantics via

$$\textsc{Abstraction}_{\textsc{cp}}(IAF, x_a) = \varphi_?(IAF) \wedge \varphi_{\textsc{cp}}(IAF) \wedge x_a.$$

If the abstraction is unsatisfiable, we *reject* the query argument. A satisfying truth assignment corresponds to a completion AF^* and a complete extension containing the query a. Then, our goal is to check whether a is skeptically accepted under preferred semantics. This is accomplished by first checking whether a is skeptically accepted in the current completion under complete semantics via

$$\textsc{Check}_{\textsc{cp}}(IAF, AF, \neg x_a) = \varphi_{\textsc{cp}}(IAF) \wedge \textsc{Completion}(IAF, AF^*) \wedge \neg x_a.$$

If this formula is unsatisfiable, we can safely *accept*, as skeptical acceptance under complete semantics implies skeptical acceptance under preferred semantics. If it is satisfiable, we get a complete extension not containing a, and enter the same subset-maximization procedure as in the algorithm for PR-NSA. Finally, we check whether including a into the subset-maximal complete extension not containing a yields a larger complete extension. If this is not possible, we refine the abstraction via $\textsc{Refine}(IAF, AF^*)$ and continue iteratively, since a is not skeptically accepted under preferred semantics in AF^*, which is witnessed by the preferred extension not containing a. If this is possible, we add the clause $\textsc{Completion}(IAF, AF^*) \rightarrow \bigvee_{b \in \mathscr{A}^* \setminus \mathscr{E}} x_b$, where $\mathscr{E}$ is the maximal complete extension, ruling out all subsets for that completion, and continue.

Finally, we remark that although our algorithm for $\textsc{ad}_{\neq\emptyset}$-PEXSA and ST-PEXSA covers the second-level incomplete variant, it also allows for solving the DP-complete problems $\textsc{ad}_{\neq\emptyset}$-EXSA and ST-EXSA. In this case, the CEGAR algorithm is bound to work within the resource limits of DP, i.e., terminating after one iteration. This is due to the fact that if we find a counterexample to the solution of the abstraction, the refinement clause is empty because there are no incomplete elements, causing immediate termination.

6.3. Strong Refinements

Recall the CEGAR algorithms for **s**-NCA (Algorithm 1) and **s**-PEXSA (Algorithm 2) for $\mathbf{s} \in \{\textsc{ad}_{\neq\emptyset}, \textsc{st}\}$. Before refining the abstraction, in both algorithms we have obtained a counterexample extension $\mathscr{E}$, which either contains (for **s**-NCA) or does not contain (for **s**-PEXSA) the query argument. In both cases, however, we would ideally like to exclude all completions which still possess the counterexample $\mathscr{E}$ as an **s** extension, since these completions would only cause more undesired iterations in the CEGAR algorithm. Due to Propositions 41–45 in Section 4, the following *strong refinement* is also valid, and excludes certain other completions which admit the counterexample extension:

$$\textsc{Refine}_{\mathbf{s}}(IAF, AF^*, \mathscr{E}) = \bigvee_{a\in\mathscr{A}^?\cap\textsc{in}(\mathscr{E})} \neg y_a \vee \bigvee_{\substack{a\in\mathscr{A}^?\setminus\mathscr{A}^*\\ (\nexists (b,a)\in\mathscr{R})[b\in\textsc{in}(\mathscr{E})]}} y_a$$
$$\vee \bigvee_{(a,b)\in\mathscr{R}^?\cap\mathscr{R}^*\cap(\textsc{in}(\mathscr{E})\times\textsc{out}(\mathscr{E}))} \neg r_{a,b} \vee \bigvee_{\substack{(a,b)\in(\mathscr{R}^?\setminus\mathscr{R}^*)\cap\\ ((\textsc{in}(\mathscr{E})\cup\textsc{undec}(\mathscr{E}))\times\textsc{in}(\mathscr{E}))}} r_{a,b}.$$

Thus in Algorithms 1 and 2 we may replace $\textsc{Refine}(IAF, AF^*)$ with $\textsc{Refine}_{\mathbf{s}}(IAF, AF^*, \mathscr{E})$ while maintaining correctness. Since in the procedure described for PR-NSA on page 30 the counterexample extension $\mathscr{E}$ is also an admissible set containing the query, here we may replace $\textsc{Refine}(IAF, AF^*) \vee \bigvee_{b\in\mathscr{A}^*\setminus\mathscr{E}} x_b$ with $\textsc{Refine}_{\textsc{ad}}(IAF, AF^*, \mathscr{E}) \vee \bigvee_{b\in\mathscr{A}^*\setminus\mathscr{E}} x_b$.

The situation is different in the procedure for PR-PSA described on page 31. Here we refine the abstraction via adding the clause $\textsc{Refine}(IAF, AF^*)$ if we find a *preferred* extension not containing the query. Thus we instead make use of Propositions 47–50 yielding the following strong refinement:

$$\textsc{Refine}_{\textsc{pr}}(IAF, AF^*, \mathscr{E}) = \bigvee_{a\in\mathscr{A}^?\cap(\textsc{in}(\mathscr{E})\cup\textsc{undec}(\mathscr{E}))} \neg y_a \vee \bigvee_{\substack{a\in\mathscr{A}^?\setminus\mathscr{A}^*\\ (\nexists (b,a)\in\mathscr{R})[b\in\textsc{in}(\mathscr{E})]}} y_a$$
$$\vee \bigvee_{\substack{(a,b)\in\mathscr{R}^?\cap\mathscr{R}^*\cap\\ ((\textsc{in}(\mathscr{E})\times\textsc{out}(\mathscr{E}))\cup(\textsc{undec}(\mathscr{E})\times\textsc{undec}(\mathscr{E})))}} \neg r_{a,b} \vee \bigvee_{\substack{(a,b)\in(\mathscr{R}^?\setminus\mathscr{R}^*)\cap\\ ((\textsc{in}(\mathscr{E})\cup\textsc{undec}(\mathscr{E}))\times(\textsc{in}(\mathscr{E})\cup\textsc{undec}(\mathscr{E})))}} r_{a,b}.$$

We will show in Section 7 that strong refinements are crucial for solving these problems in practice.

7. Experiments

We continue by an overview of results from an empirical evaluation of the scalability of the approaches to acceptance problems in incomplete AFs described in Section 6. Our implementation, TAEYDENNAE,[3] uses Glucose 4.1 [4] as the underlying SAT solver, and is available in open source:

`https://bitbucket.org/andreasniskanen/taeydennae` .

In the implementation, we employ fully incremental SAT solving, making use of the assumptions interface of Glucose.

7.1. Experiment Setup

We generated incomplete AFs based on the benchmarks used in the 2nd International Competition on Computational Models of Argumentation (ICCMA 2017) [53] as follows. For each AF, we select a query argument uniformly at random from the set of arguments. For each probability $p \in \{0.05, 0.1, 0.15, 0.2\}$, we generated three incomplete AFs: one where each argument (except for the query) is uncertain with probability p, one where each attack is uncertain with probability p, and one where each argument and attack is uncertain with probability p. We used the ICCMA 2017 benchmark set A for problems on the second level and the set B for problems on the first level, in line with the complexity of the acceptance problems for which these sets were used in ICCMA 2017. This resulted in a total of 4200 IAFs for each of the two ICCMA 2017 benchmark sets. A script for generating these IAF instances from ICCMA 2017 instances is included in the solver repository.

The experiments were run on Intel Xeon E5-2680 v4 2.4-GHz, 256-GB machines with CentOS 7. We set a per-instance time limit of 900 seconds and a per-instance memory limit of 64 GB.

[3] “Täydennä” is the imperative form of the verb “to complete” in Finnish.

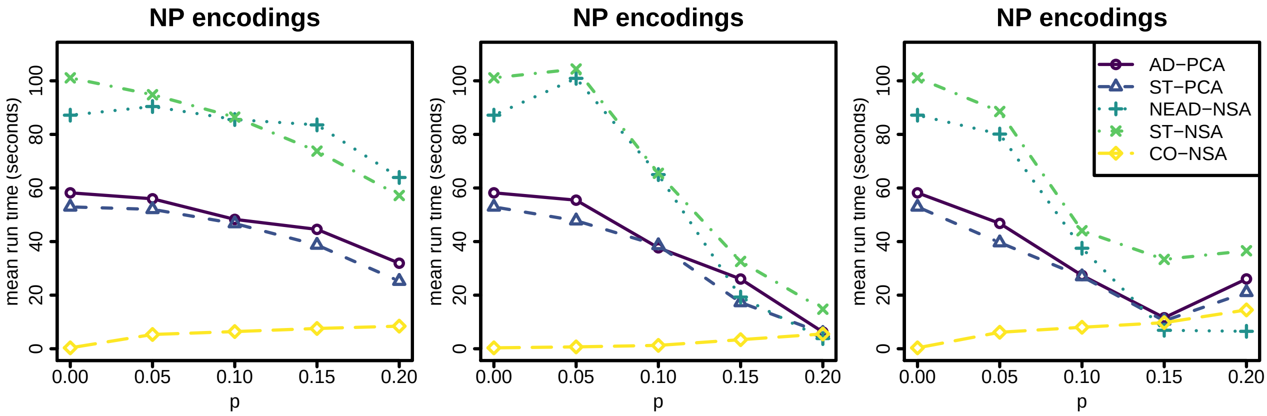


Figure 11: Mean run times (with timeouts included) for purely argument-incomplete (left), purely attack-incomplete (center), and general incomplete (right) AFs for the problems on the first level.

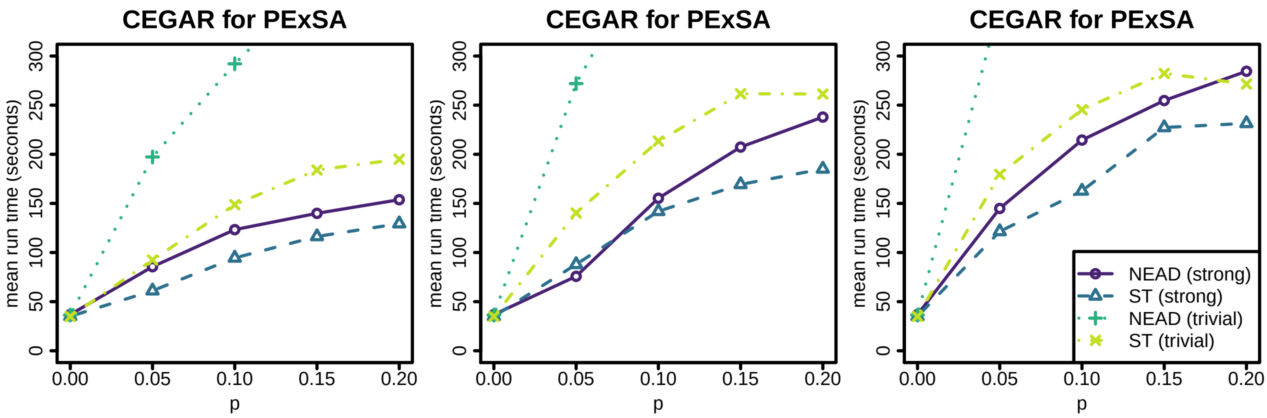


Figure 12: Mean run times (with timeouts included) for purely argument-incomplete (left), purely attack-incomplete (center), and general incomplete (right) AFs for possible existence and skeptical acceptance.

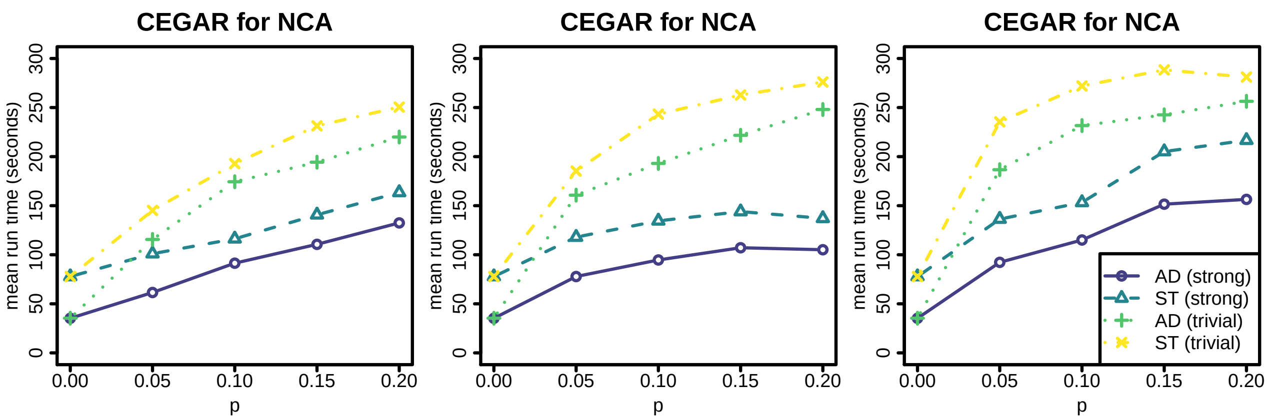


Figure 13: Mean run times (with timeouts included) for purely argument-incomplete (left), purely attack-incomplete (center), and general incomplete (right) AFs for necessary credulous acceptance.

7.2. Results

For the NP-complete problems AD-PCA (= CP-PCA = PR-PCA) and ST-PCA, and the coNP-complete problems $\text{AD}_{\neq\emptyset}$-NSA, ST-NSA, and CP-NSA (= GR-NSA = GR-NCA), the mean run times (with timeouts included as the timeout limit of 900 seconds) are visualized in Figure 11 for different values of p and argument-incompleteness or attack-incompleteness. Interestingly, the empirical hardness of the instances does not increase as incompleteness is increased; in fact, for problems other than CP-NSA, we are able to solve the corresponding incomplete instance sets faster on average than for $p = 0$, which is exactly the original ICCMA benchmark set (i.e., "normal" acceptance problems in standard AFs), especially when introducing attack-incompleteness. We hypothesize this to be due to the fact that by making certain elements uncertain, there are more possibilities to have a satisfiable SAT instance—this is witnessed by the fact that as p is increased, the number of *accept* answers increases considerably for PCA problems (e.g., from 35% of solved instances with $p = 0$ to 71% with $p = 0.2$ for AD-PCA), and decreases for NSA problems (e.g., from 56% of solved instances with $p = 0$ to 21% with $p = 0.2$ for ST-NSA). The only exception is CP-NSA which is, for our approach and for this set of instances, surprisingly fast to solve, with mean running times around 10 seconds for each choice of parameters despite coNP-completeness of the task for incomplete AFs (as opposed to polynomial-time computability for standard AFs). This can be explained by the fact that the number of *accept* answers, i.e., unsatisfiable SAT instances, is very low (about between 6% and 8%) for all choices of p.

The Σ_2^p-complete problems $\text{AD}_{\neq\emptyset}$-PExSA and ST-PExSA are solved via the CEGAR approach. The mean run times (with timeouts included as 900 seconds) are shown in Figure 12. We observe that, in contrast to the NP encodings, introducing uncertainty makes the instances significantly harder to solve. We suspect this to be due to the fact that the number of potential completions to guess and check is exponential in the number of uncertain elements. However, the strong refinements described in Section 6.3 resulting from the analysis presented in Section 4, significantly speed up the CEGAR approach. Mean run times when using the trivial refinement are considerably higher than when using the strong refinement, especially for $\text{AD}_{\neq\emptyset}$-PExSA, to the extent that the strong refinements are central in making the CEGAR approach viable for deciding acceptance in incomplete AFs. This is also evident from Figure 14 (top left and top right), which shows the running times for each instance using the trivial refinement plotted on the x-axis and for the strong refinement on the y-axis. While a major part of the instances takes almost the exact same amount of time to solve, the effect of strong refinements is witnessed by the large number of timeouts for the trivial refinement on the right-hand side of each plot for instances that are solved within the timeout limit using the strong refinement.

The Π_2^p-complete problems AD-NCA and ST-NCA are also solved employing CEGAR, with the mean running times (with timeouts included) shown in Figure 13. Here we also observe that the empirical hardness of instances increases significantly when increasing incompleteness, both in terms of increasing p and in terms of introducing both argument- and attack-incompleteness. Again, in the worst case we need to check for an exponential number of completions with respect to the number of uncertain elements, which we hypothesize to be the cause of this behavior. Employing strong refinements is essential also for these problems, as further shown in Figure 14 (bottom left and bottom right).

Finally, mean run times for the Σ_3^p-complete problem PR-PSA are shown in Figure 15 (left). We observe that, similarly to the problems complete for the second level, introducing uncertainty both via increasing p and introducing both argument- and attack-incompleteness increases the mean running times. Perhaps surprisingly, the scale of the mean run times is also similar despite the gap in theoretical complexity. Clearly, also in this case employing strong refinements is crucial for solving instances of this problem efficiently, as is evident from significantly smaller run times on average, as well as the large number of timeouts shown in Figure 15 (right).

Table 4: Number of timeouts for each problem variant using a basic enumeration approach (enum.) and direct SAT or SAT-based CEGAR (SAT).

problem	AD-PCA	ST-PCA	$\text{AD}_{\neq\emptyset}$-NSA	ST-NSA	CP-NSA	$\text{AD}_{\neq\emptyset}$-PExSA	ST-PExSA	AD-NCA	ST-NCA	PR-PSA
enum.	2631	2874	1194	2301	329	3608	3049	985	852	3001
SAT	**121**	**111**	**164**	**192**	**19**	**752**	**603**	**409**	**559**	**822**

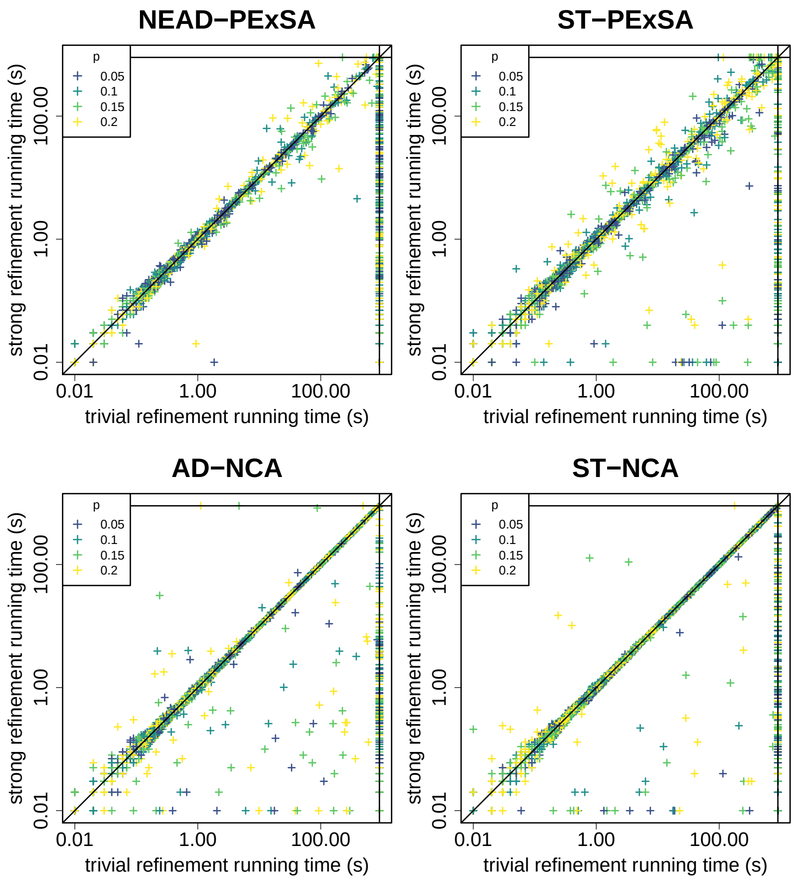


Figure 14: Trivial vs. strong refinement for $\mathrm{AD}_{\neq\emptyset}$-PExSA (top left), ST-PExSA (top right), AD-NCA (bottom left), and ST-NCA (bottom right).

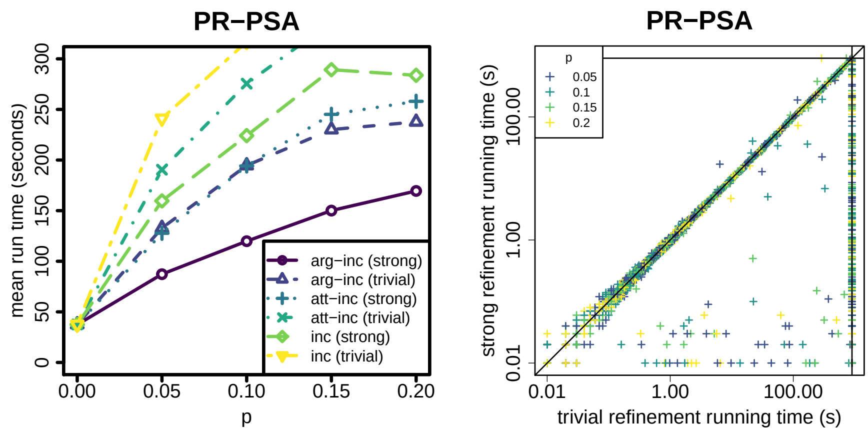


Figure 15: Mean run times (with timeouts included) for argument-incomplete, attack-incomplete, and general incomplete AFs for PR-PSA (left), trivial vs. strong refinement for PR-PSA (right).

7.3. Empirical Comparison with Enumeration-based Acceptance

The SAT-based algorithms presented in this article are the first ones proposed in the context of incomplete AFs, barring direct empirical runtimes comparison with other approaches. However, in principle the acceptance problems considered in this work can also be decided with a more simple algorithmic approach which enumerates all completions of an input IAF and, for each completion, calls a standard AF solver for the respective standard acceptance problem. For obtaining a baseline for an empirical scalability comparison, we implemented this enumeration approach based on the state-of-the-art AF solver μ–TOKSIA [68]. For a “possible” variant, the enumeration algorithm outputs “yes” as soon as the AF solver outputs “yes.” Dually, for a “necessary” variant, the algorithm outputs “no” as soon as the AF solver outputs “no.” Otherwise, it enumerates through all completions of the input IAF. If this is the case, note that, for an input IAF with a large number of uncertain elements (and thus completions), this algorithm cannot be expected to terminate in a reasonable amount of time, as the number of completions becomes simply too large.

The results of a comparison of the empirical performance of our SAT-based algorithms and the simple SAT-based enumeration approach are summarized in Table 4. For the first-level problems AD-PCA, ST-PCA, AD$_{\neq\emptyset}$-NSA, ST-NSA, and CP-NSA, the direct SAT-based approach considerably outperforms the enumeration-based approach, with the enumeration-based approach exhibiting at least seven times as many timeouts as the direct one. For the Σ_2^p-complete problems AD$_{\neq\emptyset}$-PEXSA and ST-PEXSA, the SAT-based CEGAR algorithm produces approximately five times fewer timeouts than the enumeration algorithm, and is thus again clearly the superior approach. The picture is less drastic but still clear for the Π_2^p-complete problems AD-NCA and ST-NCA, for which the enumeration-based method produces more timeouts than the direct one by a factor of 2.4 and 1.5, respectively. We hypothesize this to be due to a large ratio of “no” answers to “yes” answers. For the Σ_3^p-complete PR-PSA, the SAT-based CEGAR again clearly outperforms the enumeration algorithm which exhibits 3.7 times as many timeouts. All in all, the enumeration-based approach turned out not to be competitive with the more intricate SAT-based algorithms developed in this work.

8. Related Work

Finally, we review works related from different angles to the work presented in this article.

8.1. Instantiation of Incomplete AFs

Various formalisms have been proposed in the literature that allow to instantiate abstract argumentation frameworks from structured data—typically consisting of literals in some formal language, inference rules that relate these literals, preference relations over different parts of the data, and possibly more auxiliary information. When there is structural uncertainty about elements in the underlying data, this may translate to structural uncertainty about arguments or attacks in the abstract argumentation framework that represents it, and thus produce incomplete argumentation frameworks. We give a few examples (for formal definitions, please refer to the original papers).

- In value-based argumentation frameworks [19], when we have two arguments a and b with associated values $val(a)$ and $val(b)$, and if it is known that the preference between the two values is $val(b) \geq val(a)$, but it is not known whether $val(b) > val(a)$ or $val(b) = val(a)$, then this results in an uncertain attack (a, b).
- A similar situation can arise when using the ASPIC⁺ model [66]: Consider two rebutting arguments a and b that have two rules r_a and r_b as their respective top rules, with preference $r_b \geq r_a$, but it is unknown whether $r_b > r_a$ or $r_b = r_a$. This may result in the rebutting attack (a, b) being uncertain.
- As another example, consider an ASPIC⁺ argument a with top rule r. If there is structural uncertainty about r in the underlying knowledge base, then this can translate to a being uncertain in the resulting argumentation framework.
- The same happens when using the method of Wyner et al. [84] to instantiate argumentation frameworks: uncertainty about the existence of a rule r in the knowledge base produces an uncertain argument r in the argumentation framework representation. Wyner et al.'s method also allows uncertain attacks: Uncertainty about whether an inference rule r is strict or defeasible in the knowledge base produces an uncertain attack (c, r), where c is the abstract argument that represents the negation of r's head literal.

8.2. IAFs and Other Generalizations of AFs

First, while this work studies "possible" and "necessary" variants of the acceptance problem in IAFs, Baumeister et al. [16, 17, 18] have studied "possible" and "necessary" variants of the verification problem and Skiba et al. [79] "possible" and "necessary" variants of the existence problem in IAFs in terms of their complexity.

Some other generalizations of abstract AFs exist in the literature that also aim at modeling uncertainty or dynamics of argumentation in a similar fashion as IAFs. One of them is the model of probabilistic argumentation frameworks (PrAFs) [60]. A PrAF is an AF with an associated probability distribution both over the set of arguments and over the set of attacks. Both IAFs and PrAFs allow to represent structural uncertainty in AFs, and both share an assumption of independence, i.e., that the existence of an uncertain element is independent of that of the other uncertain elements. However, PrAFs contain more information than IAFs, which, as a positive consequence, allows to represent uncertainty more precisely when there is detailed information about probabilities. On the other hand, they overdefine cases where the likelihood of existence of the uncertain elements is not known exactly. When this difference is disregarded, PrAFs and IAFs can be mapped to each other by the following procedure: For any IAF, a corresponding probabilistic argumentation framework *PrAF* can be obtained by setting the probability $P_{PrAF}(x) = 1$ for all definite elements $x \in \mathscr{A} \cup \mathscr{R}$ and setting $0 < P_{PrAF}(y) < 1$ for all uncertain elements $y \in \mathscr{A}^? \cup \mathscr{R}^?$. However, this representation indicates a level of precision that may not be justified, since the incomplete AF does not provide information about how likely the existence of its uncertain elements is. Conversely, any probabilistic argumentation framework *PrAF* can be represented as an IAF by including all arguments and attacks x with $P_{PrAF}(x) = 1$ as definite elements, and all elements y with $0 < P_{PrAF}(y) < 1$ as uncertain elements. This is a lossy representation, since the different probabilities of the uncertain elements are not preserved. Given any pair of *IAF* and *PrAF* corresponding to each other, the possible verification problem **s**-INCPV[4] for *IAF* corresponds to the question of whether the target set of arguments has probability greater than zero to be an **s** extension of *PrAF*, and the necessary verification problem **s**-INCNV[5] for *IAF* corresponds to the question of whether the target set of arguments has probability 1 to be an **s** extension of *PrAF*:

$$
\begin{aligned}
(IAF, \mathscr{E}) \in \mathbf{s}\text{-}\textsc{IncPV} \quad &\Leftrightarrow \quad P^{\mathbf{s}}_{PrAF}(\mathscr{E}) > 0, \\
(IAF, \mathscr{E}) \in \mathbf{s}\text{-}\textsc{IncNV} \quad &\Leftrightarrow \quad P^{\mathbf{s}}_{PrAF}(\mathscr{E}) = 1.
\end{aligned}
$$

Similarly, the generalized acceptance problems for incomplete AFs studied in this article correspond to the generalizations $\text{P-CA}^{\mathbf{s}}_{PrAF}$ of **s**-CA and $\text{P-SA}^{\mathbf{s}}_{PrAF}$ of **s**-SA for probabilistic AFs, which are functional problems that provide the probability for a single argument to be credulously accepted in *PrAF* ($\text{P-CA}^{\mathbf{s}}_{PrAF}$), respectively, to be skeptically accepted in *PrAF* ($\text{P-SA}^{\mathbf{s}}_{PrAF}$), using semantics **s**. The correpondences are as follows:

$$
\begin{aligned}
(IAF, a) \in \mathbf{s}\text{-PCA} \quad &\Leftrightarrow \quad \text{P-CA}^{\mathbf{s}}_{PrAF}(a) > 0, \\
(IAF, a) \in \mathbf{s}\text{-NCA} \quad &\Leftrightarrow \quad \text{P-CA}^{\mathbf{s}}_{PrAF}(a) = 1, \\
(IAF, a) \in \mathbf{s}\text{-PSA} \quad &\Leftrightarrow \quad \text{P-SA}^{\mathbf{s}}_{PrAF}(a) > 0, \\
(IAF, a) \in \mathbf{s}\text{-NSA} \quad &\Leftrightarrow \quad \text{P-SA}^{\mathbf{s}}_{PrAF}(a) = 1.
\end{aligned}
$$

However, the expressive power of PrAFs compared to IAFs comes at a computational cost: Fazzinga et al. [52] show that the $\textsc{Prob}^{\mathbf{s}}_{PrAF}$ problem—which is the generalization of the **s**-VERIFICATION problem for PrAFs—is $\mathrm{FP}^{\#P}$-complete for all semantics considered here except the conflict-free, admissible, and stable semantics. Similarly, Fazzinga et al. [51] show that the $\text{P-CA}^{\mathbf{s}}_{PrAF}$ and $\text{P-SA}^{\mathbf{s}}_{PrAF}$ problems are $\mathrm{FP}^{\#P}$-complete, too, for every nontrivial semantics or property. $\mathrm{FP}^{\#P}$-complete problems are deemed intractable in practice—after all, by Toda's theorem [81], every problem of the polynomial hierarchy can be solved by a polynomial-time Turing-reduction to a $\#P$ oracle. Compared to this, all verification and acceptance problem variants for IAFs are within the first three levels of the polynomial hierarchy, with the necessary verification problem **s**-INCNV for IAFs and the possible verification problem **s**-ATTINCPV for attack-incomplete AFs even being in P for most semantics [17].

The model of incomplete argumentation frameworks is further closely related to the recently proposed *control argumentation frameworks* (CAF) [36], which use a similar, yet more specific model of uncertainty in argumentation

[4] In **s**-INCPV, we are given an incomplete argumentation framework $IAF = \langle \mathscr{A}, \mathscr{A}^?, \mathscr{R}, \mathscr{R}^? \rangle$ and a set $S \subseteq \mathscr{A} \cup \mathscr{A}^?$, and the question is whether there exists a completion $AF^* = \langle \mathscr{A}^*, \mathscr{R}^* \rangle$ such that $S|_{\mathscr{A}^*} = S \cap \mathscr{A}^*$ is an **s** extension of AF^*.

[5] **s**-INCNV is defined similarly to **s**-INCPV, with the distinction of quantifying universally over completions.

frameworks that is specifically aimed at representing strategic scenarios, and find applications in, e.g., argument-based negotiation [37]. There are various cases where both models coincide. For example, *possible* problem variants in purely argument-incomplete argumentation frameworks can be represented by CAFs using their *control*-part, while *necessary* problem variants in incomplete argumentation frameworks can be represented by CAFs using their *uncertain*-part. Although CAFs include an additional symmetric attack relation in the uncertain part, where only the direction—not the existence—of an attack is uncertain, the complexity results of this article also hold for incomplete AFs augmented with this form of uncertainty. Furthermore, the SAT encodings presented in this article are also easily adapted to this formalism. In fact, the results of this work have subsequently (after the writing of this article) proven to be useful for the complexity analysis of and algorithms for the so-called controllability problem in CAFs [70, 64].

8.3. Dynamics of Argumentation Frameworks

The PCA problem in incomplete argumentation frameworks is related to extension enforcement [9, 35], where, given an argumentation framework and a subset of its arguments, the task is to determine how the attack relation and/or the set of arguments of the argumentation framework can be modified in a smallest possible way so that the given set becomes part of an extension. Instances for acceptance problems in incomplete argumentation frameworks and for enforcement problems coincide when the incomplete argumentation framework has only uncertain attacks and no uncertain arguments, and when the enforcement instance allows only changes to the attack relation and its given subset is a singleton. However, enforcement aims at finding a smallest possible change to the argumentation framework, which is not the aim in deciding acceptance in incomplete argumentation frameworks. On the other hand, the question of whether acceptance of a target argument can *at all* be achieved is trivially true in most variants of enforcement, while this is the key question for possible-credulous acceptance problems in incomplete argumentation frameworks. Furthermore, recent work on maximizing goals achievement (in terms of argument labels) while minimizing the number of actions (additions and deletions of arguments) [32] is also related to the PCA problem in argument-incomplete argumentation frameworks. While in PCA we look at the acceptance of a specific target argument, which can be seen as a single IN goal, the aforementioned problem has several target arguments expressed as both IN and OUT goals. A key difference is also that we do not consider optimizing over the presence or absence of uncertain arguments, which can be seen as actions in the aforementioned problem.

Closely related to the analysis presented in Section 4, Cayrol et al. [28] study the problem of adding an argument and incident attacks, specifically by studying necessary and sufficient conditions for satisfying different properties, also defining the atomic changes (adding and removing arguments and attacks) we study in the context of incomplete AFs. However, their focus is on grounded and preferred semantics, whereas we consider admissible and stable semantics. Semantical change when removing an argument along with incident attacks was studied by Bisquert et al. [22] under preferred, stable, and grounded semantics, also focusing on the satisfaction of properties of extensions. The preservation of the grounded extension was studied by Boella et al. [23, 24] when removing arguments and attacks, or adding attacks; again, we focus on admissible and stable extensions. Finally, the work of Rienstra et al. [77] focuses on the preservation of grounded, complete, preferred, stable, and semi-stable labelings under changes to the attack structure, and their results for the stable and preferred semantics coincide with ours.

Incremental algorithms for dynamic argumentation frameworks, where in addition to an AF a change or a sequence of changes to the attack structure is provided, and the task is to answer a query for all AFs defined by the sequence of changes, have been recently studied both in the context of computing extensions [54, 55, 1] and for acceptance problems [2]. These algorithms build on the division-based method [61, 7] which divides the updated AF into affected and unaffected parts, and expand this method by, e.g., taking into account an initial extension of the first AF. In dynamic AFs, the sequence of changes is provided as input, hence the number of AFs considered is linear with respect to the size of the input. In contrast, in incomplete AFs the number of completions is exponential with respect to the input size. However, via casting a dynamic AF to an attack-incomplete AF by setting all attacks amenable to change as uncertain attacks, a "no" answer to **s**-PCA implies that the answer is "no" also for all AFs defined by the dynamic AF. Similarly, a "yes" answer to **s**-NSA implies that the answer is "yes" for all AFs of the dynamic AF. In a sense, this work on reasoning in incomplete AFs provides a shortcut for acceptance problems in dynamic AFs.

The problem of adding an argument along with incident attacks is also related to expansions [11, 9, 8], where sets of new arguments along with incident attacks are added to an AF. Likewise, removing arguments has been studied in the form of deletions [10]. There may be potential for using further theoretical results of expansion, deletion, and update equivalence in order to strengthen the refinement in the CEGAR algorithm. However, equivalence is a considerably more general concept, as it concerns preserving all extensions of a given AF, whereas in our approach we are interested in preserving just the counterexample. In addition, recent work on preprocessing argumentation frameworks via so-called replacement patterns [45], based on a parameterized notion of equivalence [12], may prove to be useful in the context of algorithms for incomplete AFs, as well. In particular, it would be interesting to study an adaptation of this form of equivalence for incomplete AFs, where changes are allowed only on the uncertain part.

8.4. SAT-Based Approaches to Argumentation

In terms of practical systems for reasoning in argumentation frameworks, the SAT-based approaches developed in this work continue the successful line of work on applying SAT-based approaches to reasoning over various computational models of argumentation. These include argument acceptance problems and extension enumeration in standard AFs [46, 58, 50, 29], other generalization of AFs such as abstract dialectical frameworks [62], as well as various forms of optimization problems underlying argumentations dynamics, including enforcement, adjustment, repairment, synthesis, reasoning in dynamic AFs, and goals achievement [83, 72, 73, 59, 74, 67, 32]. The strong refinements we develop have subsequently (after the writing of this article) shown to be applicable in the context of CEGAR algorithms for second-level complete variants of enforcement and synthesis [69]. Similar ideas have also been considered in the specific context of the NP-complete problem of extension enforcement under grounded semantics [73].

9. Conclusion

The recently proposed notion of incomplete argumentation frameworks generalizes Dung's standard abstract argumentation frameworks by allowing for modeling uncertain attacks and arguments. In contrast to standard AFs, the computational complexity of variants of acceptance problems in the context of incomplete AFs has not been thoroughly established to-date. Furthermore, the introduction of incomplete AFs raises the challenge of developing practical decision procedures for reasoning about acceptance under uncertainties. In this article, we address both of these challenges. In particular, we proposed natural generalizations of credulous and skeptical acceptance in AFs to incomplete AFs, giving rise to several variants for both of the two reasoning modes.

By establishing a full complexity landscape of acceptance in AFs for the variants under various central argumentation semantics (see Table 1 on page 9), we showed that acceptance in incomplete AFs is most often hard for the first level of the polynomial hierarchy, and can reach completeness for this hierarchy's second or even third level in some cases. Motivated by the success of SAT-based practical decision procedures developed for reasoning in standard AFs, we proposed SAT-based algorithms for all of the variants of acceptance in incomplete AFs covered by our complexity analysis. We showed through an empirical evaluation that for NP-complete variants of acceptance, reasoning in incomplete AFs turns out to be at least as efficient in practice as reasoning about acceptance in standard AFs, and that the CEGAR approaches we developed for the variants with beyond-NP complexity also scale to instances of reasonable size.

While the complexity results provided in this article cover several central argumentation semantics, the analysis could be extended to cover even further semantics such as ideal [40], semi-stable [27], and stage [82] semantics. Exploring SCC-recursive semantics [6] in the context of incomplete AFs is also a possible interesting direction for future work. Analyzing the complexity of acceptance in structural or distance-based subclasses of incomplete AFs would also be of interest, in analogy with related analyses provided earlier in the context of standard AFs [34, 42, 48, 47, 46]. Further, it would also be interesting to extend both the complexity analysis and the decision procedures presented in this article to other related formalisms that allow to represent unquantified uncertainty, such as control argumentation frameworks. In terms of potential improvements to the empirical performance of the proposed decision procedures, further analysis on possible ways of obtaining even stronger refinements than those obtained through the presented analysis on the persistence of extensions under change may be fruitful.

Acknowledgements

We thank the journal reviewers as well as the reviewers of the preliminary conference version reviewers for helpful suggestions. This work has been financially supported in part by Academy of Finland (grants 276412, 312662, and

322869), by University of Helsinki Doctoral Programme in Computer Science DoCS, by Deutsche Forschungsgemeinschaft under DFG grants RO 1202/14-2 and RO 1202/21-1, and by the project "Online Participation" funded by the NRW Ministry for Innovation, Science, and Research.

References

[1] G. Alfano, S. Greco, and F. Parisi. Efficient computation of extensions for dynamic abstract argumentation frameworks: An incremental approach. In *Proc. IJCAI'17*, pages 49–55. ijcai.org, 2017.

[2] G. Alfano, S. Greco, and F. Parisi. An efficient algorithm for skeptical preferred acceptance in dynamic argumentation frameworks. In *Proc. IJCAI'19*, pages 18–24. ijcai.org, 2019.

[3] E. and A. Hunter. A relevance-theoretic framework for constructing and deconstructing enthymemes. *Journal of Logic and Computation*, 22(1):55–78, 2012.

[4] G. Audemard and L. Simon. On the Glucose SAT solver. *International Journal on Artificial Intelligence Tools*, 27(1):1–25, 2018.

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

[6] P. Baroni, M. Giacomin, and G. Guida. Scc-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence*, 168(1-2):162–210, 2005.

[7] P. Baroni, M. Giacomin, and B. Liao. On topology-related properties of abstract argumentation semantics. A correction and extension to dynamics of argumentation systems: A division-based method. *Artificial Intelligence*, 212:104–115, 2014.

[8] R. Baumann. Normal and strong expansion equivalence for argumentation frameworks. *Artificial Intelligence*, 193:18–44, 2012.

[9] R. Baumann. What does it take to enforce an argument? Minimal change in abstract argumentation. In *Proc. ECAI'12*, volume 242 of *FAIA*, pages 127–132. IOS Press, 2012.

[10] R. Baumann. Context-free and context-sensitive kernels: Update and deletion equivalence in abstract argumentation. In *Proc. ECAI'14*, volume 263 of *FAIA*, pages 63–68. IOS Press, 2014.

[11] R. Baumann and G. Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In *Proc. COMMA'10*, volume 216 of *FAIA*, pages 75–86. IOS Press, 2010.

[12] R. Baumann, W. Dvořák, T. Linsbichler, and S. Woltran. A general notion of equivalence for abstract argumentation. *Artificial Intelligence*, 275:379–410, 2019.

[13] D. Baumeister, D. Neugebauer, and J. Rothe. Collective acceptability in abstract argumentation. In D. Gabbay, M. Giacomin, G. Simari, and M. Thimm, editors, *Handbook of Formal Argumentation*, volume 2. College Publications. To appear.

[14] D. Baumeister, D. Neugebauer, and J. Rothe. Verification in attack-incomplete argumentation frameworks. In *Proc. ADT'15*, volume 9346 of *LNAI*, pages 341–358. Springer, 2015.

[15] D. Baumeister, D. Neugebauer, and J. Rothe. Credulous and skeptical acceptance in incomplete argumentation frameworks. In *Proc. COMMA'18*, volume 305 of *FAIA*, pages 181–192. IOS Press, 2018.

[16] D. Baumeister, D. Neugebauer, J. Rothe, and H. Schadrack. Complexity of verification in incomplete argumentation frameworks. In *Proc. AAAI'18*, pages 1753–1760. AAAI Press, 2018.

[17] D. Baumeister, D. Neugebauer, J. Rothe, and H. Schadrack. Verification in incomplete argumentation frameworks. *Artificial Intelligence*, 264:1–26, 2018.

[18] D. Baumeister, J. Rothe, and H. Schadrack. Verification in argument-incomplete argumentation frameworks. In *Proc. ADT'15*, volume 9346 of *LNAI*, pages 359–376. Springer, 2015.

[19] T. J. M. Bench-Capon. Value-based argumentation frameworks. In *Proc. NMR'02*, pages 444–453, 2002.

[20] P. Besnard and S. Doutre. Checking the acceptability of a set of arguments. In *Proc. NMR'04*, pages 59–64, 2004.

[21] A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors. *Handbook of Satisfiability*, volume 185 of *FAIA*. IOS Press, 2009.

[22] P. Bisquert, C. Cayrol, F. Dupin de Saint-Cyr, and M. Lagasquie-Schiex. Change in argumentation systems: Exploring the interest of removing an argument. In *Proc. SUM'11*, volume 6929 of *LNCS*, pages 275–288. Springer, 2011.

[23] G. Boella, S. Kaci, and L. van der Torre. Dynamics in argumentation with single extensions: Abstraction principles and the grounded extension. In *Proc. ECSQARU'09*, volume 5590 of *LNAI*, pages 107–118. Springer, 2009.

[24] G. Boella, S. Kaci, and L. van der Torre. Dynamics in argumentation with single extensions: Attack refinement and the grounded extension (extended version). In *Proc. ArgMAS'10*, volume 6057 of *LNCS*, pages 150–159. Springer, 2010.

[25] J. Cai, T. Gundermann, J. Hartmanis, L. Hemachandra, V. Sewelson, K. Wagner, and G. Wechsung. The boolean hierarchy I: Structural properties. *SIAM Journal on Computing*, 17(6):1232–1252, 1988.

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

[27] M. Caminada. Semi-stable semantics. In *Proc. COMMA'06*, volume 144 of *FAIA*, pages 121–130. IOS Press, 2006.

[28] C. Cayrol, F. de Saint-Cyr, and M. Lagasquie-Schiex. Change in abstract argumentation frameworks: Adding an argument. *Journal of Artificial Intelligence Research*, 38:49–84, 2010.

[29] F. Cerutti, M. Giacomin, and M. Vallati. How we designed winning algorithms for abstract argumentation and which insight we attained. *Artificial Intelligence*, 276:1–40, 2019.

[30] E. M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. Counterexample-guided abstraction refinement for symbolic model checking. *Journal of the ACM*, 50(5):752–794, 2003.

[31] E. M. Clarke, A. Gupta, and O. Strichman. SAT-based counterexample-guided abstraction refinement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(7):1113–1123, 2004.

[32] A. Cohen, S. Gottifredi, M. Vallati, A.J. García, and G. Antoniou. Maximising goals achievement through abstract argumentation frameworks: An optimal approach. *Expert Systems with Applications*, 141, 2020.

[33] S. Coste-Marquis, C. Devred, S. Konieczny, M. Lagasquie-Schiex, and P. Marquis. On the merging of Dung's argumentation systems. *Artificial Intelligence*, 171(10):730–753, 2007.

[34] S. Coste-Marquis, C. Devred, and P. Marquis. Symmetric argumentation frameworks. In *Proc. ECSQARU'05*, volume 3571 of *LNAI*, pages 317–328. Springer, 2005.

[35] S. Coste-Marquis, S. Konieczny, J. Mailly, and P. Marquis. Extension enforcement in abstract argumentation as an optimization problem. In *Proc. IJCAI'15*, pages 2876–2882. AAAI Press/IJCAI, 2015.

[36] Y. Dimopoulos, J. Mailly, and P. Moraitis. Control argumentation frameworks. In *Proc. AAAI'18*, pages 4678–4685. AAAI Press, 2018.

[37] Y. Dimopoulos, J. Mailly, and P. Moraitis. Argumentation-based negotiation with incomplete opponent profiles. In *Proc. AAMAS'19*, pages 1252–1260. IFAAMAS, 2019.

[38] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science*, 170(1):209–244, 1996.

[39] S. Doutre and J. Mailly. Constraints and changes: A survey of abstract argumentation dynamics. *Argument & Computation*, 9(3):223–248, 2018.

[40] P. Dung, P. Mancarella, and F. Toni. A dialectic procedure for skeptical, assumption-based argumentation. In *Proc. COMMA'06*, volume 144 of *FAIA*, pages 145–156. IOS Press, 2006.

[41] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and *n*-person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[42] P. E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artificial Intelligence*, 171(10–15):701–729, 2007.

[43] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(1):187–203, 2002.

[44] P. E. Dunne and M. Wooldridge. Complexity of abstract argumentation. In I. Rahwan and G. Simari, editors, *Argumentation in Artificial Intelligence*, chapter 5, pages 85–104. Springer, 2009.

[45] W. Dvořák, M. Järvisalo, T. Linsbichler, A. Niskanen, and S. Woltran. Preprocessing argumentation frameworks via replacement patterns. In *Proc. JELIA'19*, volume 11468 of *LNCS*, pages 116–132. Springer, 2019.

[46] W. Dvořák, M. Järvisalo, J. P. Wallner, and S. Woltran. Complexity-sensitive decision procedures for abstract argumentation. *Artificial Intelligence*, 206:53–78, 2014.

[47] W. Dvořák, S. Ordyniak, and S. Szeider. Augmenting tractable fragments of abstract argumentation. *Artificial Intelligence*, 186:157–173, 2012.

[48] W. Dvořák, R. Pichler, and S. Woltran. Towards fixed-parameter tractable algorithms for abstract argumentation. *Artificial Intelligence*, 186:1–37, 2012.

[49] W. Dvořák and S. Woltran. On the intertranslatability of argumentation semantics. *Journal of Artificial Intelligence Research*, 41:445–475, 2011.

[50] W. Faber, M. Vallati, F. Cerutti, and M. Giacomin. Solving set optimization problems by cardinality optimization with an application to argumentation. In *Proc. ECAI'16*, volume 285 of *FAIA*, pages 966–973. IOS Press, 2016.

[51] B. Fazzinga, S. Flesca, and F. Furfaro. Credulous and skeptical acceptability in probabilistic abstract argumentation: complexity results. *Intelligenza Artificiale*, 12(2):181–191, 2018.

[52] B. Fazzinga, S. Flesca, and F. Parisi. On the complexity of probabilistic abstract argumentation frameworks. *ACM Transactions on Computational Logic*, 16(3):22:1–22:39, 2015.

[53] S.A. Gaggl, T. Linsbichler, M. Maratea, and S. Woltran. Design and results of the second international competition on computational models of argumentation. *Artificial Intelligence*, 279, 2020.

[54] S. Greco and F. Parisi. Efficient computation of deterministic extensions for dynamic abstract argumentation frameworks. In *Proc. ECAI'16*, volume 285 of *FAIA*, pages 1668–1669. IOS Press, 2016.

[55] S. Greco and F. Parisi. Incremental computation of deterministic extensions for dynamic argumentation frameworks. In *Proc. JELIA'16*, volume 10021 of *LNCS*, pages 288–304, 2016.

[56] A. Hunter. Real arguments are approximate arguments. In *Proc. AAAI'07*, pages 66–71. AAAI Press, 2007.

[57] J. Köbler, U. Schöning, and K. Wagner. The difference and truth-table hierarchies for NP. *R.A.I.R.O. Informatique théorique et Applications*, 21:419–435, 1987.

[58] J. Lagniez, E. Lonca, and J. Mailly. CoQuiAAS: A constraint-based quick abstract argumentation solver. In *Proc. ICTAI'15*, pages 928–935. IEEE Computer Society, 2015.

[59] T. Lehtonen, A. Niskanen, and M. Järvisalo. SAT-based approaches to adjusting, repairing, and computing largest extensions of argumentation frameworks. In *Proc. COMMA'18*, volume 305 of *FAIA*, pages 193–204. IOS Press, 2018.

[60] H. Li, N. Oren, and T. Norman. Probabilistic argumentation frameworks. In *Proc. TAFA'11*, volume 7132 of *LNAI*, pages 1–16. Springer, July 2011.

[61] B. Liao, L. Jin, and R.C. Koons. Dynamics of argumentation systems: A division-based method. *Artificial Intelligence*, 175(11):1790–1814, 2011.

[62] T. Linsbichler, M. Maratea, A. Niskanen, J. P. Wallner, and S. Woltran. Novel algorithms for abstract dialectical frameworks based on complexity analysis of subclasses and SAT solving. In *Proc. IJCAI'18*, pages 1905–1911. AAAI Press/IJCAI, 2018.

[63] J. Mailly. Using enthymemes to fill the gap between logical argumentation and revision of abstract argumentation frameworks. *CoRR*, abs/1603.08789, 2016.

[64] J. Mailly. Possible controllability of control argumentation frameworks. In *Proc. COMMA'20*. IOS Press, 2020.

[65] A. Meyer and L. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proc. FOCS'72*, pages 125–129. IEEE Press, 1972.

[66] S. Modgil and H. Prakken. The ASPIC$^+$ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.

[67] A. Niskanen and M. Järvisalo. Algorithms for dynamic argumentation frameworks: An incremental SAT-based approach. In *Proc. ECAI'20*, volume 325 of *FAIA*, pages 849–856. IOS Press, 2020.

[68] A. Niskanen and M. Järvisalo. μ–toksia: An efficient abstract argumentation reasoner. In *Proc. KR*, pages 800–804. IJCAI, 2020.

[69] A. Niskanen and M. Järvisalo. Strong refinements for hard problems in argumentation dynamics. In *Proc. ECAI'20*, volume 325 of *FAIA*, pages 841–848. IOS Press, 2020.

[70] A. Niskanen, D. Neugebauer, and M. Järvisalo. Controllability of control argumentation frameworks. In *Proc. IJCAI-PRICAI'20*, pages 1855–1861. ijcai.org, 2020.

[71] A. Niskanen, D. Neugebauer, M. Järvisalo, and J. Rothe. Deciding acceptance in incomplete argumentation frameworks. In *Proc. AAAI'20*, pages 2942–2949. AAAI Press, 2020.

[72] A. Niskanen, J. P. Wallner, and M. Järvisalo. Optimal status enforcement in abstract argumentation. In *Proc. IJCAI'16*, pages 1216–1222. AAAI Press/IJCAI, 2016.

[73] A. Niskanen, J. P. Wallner, and M. Järvisalo. Extension enforcement under grounded semantics in abstract argumentation. In *Proc. KR'18*, pages 178–183. AAAI Press, 2018.

[74] A. Niskanen, J. P. Wallner, and M. Järvisalo. Synthesizing argumentation frameworks from examples. *Journal of Artificial Intelligence Research*, 66:503–554, 2019.

[75] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 2nd edition, 1995. Reprinted with corrections.

[76] C. Papadimitriou and M. Yannakakis. The complexity of facets (and some facets of complexity). *Journal of Computer and System Sciences*, 28(2):244–259, 1984.

[77] T. Rienstra, C. Sakama, and L. van der Torre. Persistence and monotony properties of argumentation semantics. In *Proc. TAFA'15*, volume 9524 of *LNCS*, pages 211–225. Springer, 2015.

[78] J. Rothe. *Complexity Theory and Cryptology. An Introduction to Cryptocomplexity*. EATCS Texts in Theoretical Computer Science. Springer-Verlag, 2005.

[79] K. Skiba, D. Neugebauer, and J. Rothe. Complexity of possible and necessary existence problems in abstract argumentation. In *Proc. ECAI'20*, volume 325 of *FAIA*, pages 897–904. IOS Press, 2020.

[80] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.

[81] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991.

[82] B. Verheij. Two approaches to dialectical argumentation: Admissible sets and argumentation stages. In *Proc. 8th Dutch Conference on Artificial Intelligence*, pages 357–368, 1996.

[83] J. P. Wallner, A. Niskanen, and M. Järvisalo. Complexity results and algorithms for extension enforcement in abstract argumentation. *Journal of Artificial Intelligence Research*, 60:1–40, 2017.

[84] A. Wyner, T. J. M. Bench-Capon, and P. E. Dunne. On the instantiation of knowledge bases in abstract argumentation frameworks. In *Proc. CLIMA'13*, pages 34–50. Springer, 2013.