THEME ARTICLE: SPECIAL ISSUE ON ARGUMENTATION-BASED REASONING

Complexity of Nonempty Existence Problems in Incomplete Argumentation Frameworks

Kenneth Skiba, *University of Koblenz-Landau, Koblenz 56070, Germany*

Daniel Neugebauer and Jörg Rothe, *Heinrich-Heine-Universität Düsseldorf, Düsseldorf 40225, Germany*

Abstract argumentation frameworks (AFs) are a prevailing model for the formal representation of argumentations in AI research. The generalized model of* incomplete *AFs extends the basic model by allowing for the representation of unquantified uncertainty about the existence of elements in an argumentation. For this extended model of AFs, we formally define a natural generalization of the nonempty existence problem, which asks whether there exists a nonempty set of arguments satisfying the conditions specified by a given semantics. Focusing on the fundamental semantics for incomplete AFs and considering a possible and a necessary variant of the nonempty existence problem, this yields a family of related problems, and we provide a full analysis of their computational complexity.

Alice (A) and Bob (B) want to buy a house together. They look at two houses and are now discussing which of them they should buy, weighing up their advantages against their disadvantages.

A: *I liked the big garden in the second house. It would be great for having a barbecue with our friends.*
B: *The garden is great but also a lot of work. I like the location of the first house, so I could ride my bike to work.*
A: *Yes, that would be good, but did you see the mold in the living room? That would be very unhealthy for us, or expensive to get rid of.*
B: *Nope, I did not see that. I was just looking at the giant flat-screen TV set we would have there.*
A: *You do realize that the previous owners will take their stuff with them when they leave? So there will not be a TV set in this house when we move in. And the mold in the living room is really a problem.*
B: *Well, I have actually read that mold spores help to have a quite healthy environment: They kill off all sorts of microbes, including really dangerous human pathogenic viruses.*
A: *What? That is nonsense. Where would you get that from?*
B: *From the Internet, so it must be true.*

Discussions like this can be represented using the well-established model of *abstract argumentation frameworks* (AF) introduced by Dung.[1] In this model, abstracting away from the actual contents of arguments, a discussion is illustrated by a directed graph whose vertices represent the arguments and whose edges display the attack relation among the arguments. However, Dung's model also has a few shortcomings. For example, it cannot handle incomplete information, like the argument *mold* from Alice that Bob was not aware of. But not only arguments, also attacks can be uncertain due to incomplete information. For instance, Bob believes that no attack against buying his favorite house is coming from the argument *mold*. Arguments and attacks that are not uncertain are said to be *definite*. Whether or not a definite

1541-1672 © 2021 IEEE
Digital Object Identifier 10.1109/MIS.2020.3046782
Date of publication 23 December 2020; date of current version 14 May 2021.

argument attacks another definite argument very much depends on the agents' individual beliefs, e.g., regarding causality of events. Lifting this from a private dispute among a couple to a matter of public affairs that is crucial for mankind and fiercely discussed these days, let us consider the abatement of carbon emissions. While people might agree that there is global warming, they may have different views on what is causing global warming (i.e., whether it is human induced or not): While some might view "global warming" attacking the argument that "due to increasing energy demand there is a need to build more coal-fired power plants," others might outright deny the existence of that attack.

Scenarios like these call for suitable extensions of the model of abstract AFs that allow the representation of structural uncertainties. One such extended model are the *incomplete AFs* due to Baumeister *et al.*[2] The idea of unquantified uncertainty underlying their model was first studied by Coste-Marquis *et al.*[3] for uncertain attacks. Later on, Baumeister *et al.*[4] studied abstract argumentation with uncertain arguments and with uncertain attacks.[5] They then generalized both ideas into one unified model of incomplete abstract AFs.[2] An *incomplete AF* represents a set of possibilities—we can either include an uncertain element in our system or leave it out of it, which is captured by the incomplete AF's *completions*.

Reasoning tasks in AFs are formalized by different decision problems for the verification of, membership in, or existence of sets of acceptable arguments (called *extensions*). An overview of the computational complexity of these problems can be found in the handbook chapter by Dvořák and Dunne.[6] The reasoning problems of *verification* (which refers to acceptability of a given set of arguments with respect to some semantics) and of either *credulous* or *skeptical acceptance* (both referring to acceptability of a single given argument with respect to some semantics) have already been generalized for incomplete AFs by Baumeister *et al.* [2,7,](On a related note, the problem of *controllability* in control AFs[8] generalizes the credulous and skeptical acceptance problems and asks whether there exists a selection of arguments in the control part (i.e., a *control configuration*), such that for all selections of elements in the uncertain part (i.e., all *completions*), a given target set of arguments is a subset of an extension. This problem was analyzed by Dimopoulos *et al.*[8] and Niskanen *et al.*[9]) Fazzinga *et al.*[10] pointed out a problematic behavior of the generalizations of the verification problem defined for IAFs by Baumeister *et al.*[4] where a set of arguments may be viewed as an extension in a completion even though not all its members exist in that completion. As the more recent work of Baumeister *et al.* [7] in this paper, we require all arguments of an extension to be present in a completion, avoiding this problematic behavior.

Completing these works, we tackle natural generalizations of the *existence problem* and the *nonempty existence* (or, *nonemptiness*, for short) *problem* of semantics for incomplete AFs. The (nonempty) existence problem refers to the question of whether, given an AF and a semantics, there exists an acceptable (or, a nonempty acceptable) set of arguments with respect to the given semantics in the first place. Even this fundamental question is hard to answer for some semantics (see Table 1 on page 9 for an overview of our complexity results). On the other hand, the existence problem is trivial when a semantics guarantees existence of at least one extension. In particular, this is the case for all semantics that allow the empty set to be an extension. For semantics that do not allow empty extensions, the existence and nonemptiness problems coincide. This is why we will focus only on the nonemptiness problem in our further analysis. Existence problems are the foundation of reasoning in abstract argumentation, since all other reasoning problems are based on the existence of extensions. Knowing the exact computational complexity of the (nonempty) existence problem can improve our understanding of the hardness of more advanced reasoning tasks—in particular, it lets us know whether the hardness of a variant of the verification or acceptance problem stems from the problem itself or whether the hardness is inherited from the underlying existence problem. Thus, an analysis of existence problems for incomplete AFs is sorely missing and crucially needed.

For incomplete AFs, following previous work by Baumeister *et al.*[2,7] and Niskanen *et al.*[11] on *verification* and *acceptance* problems, we look at *possible* and *necessary* variations of the existence problem as follows.

- In the *possible* problem variant, we ask whether *there exists* some completion of a given incomplete AF such that there is a nonempty set of arguments satisfying the condition of a given semantics.
- In the *necessary* problem variant, we ask whether *for every* completion of a given incomplete AF, there exists a nonempty set of arguments satisfying a given semantics.

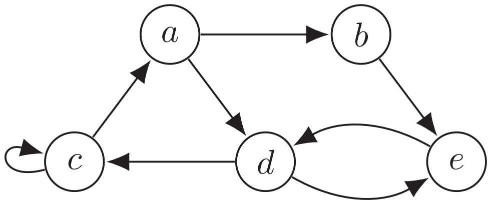

FIGURE 1. Graph representation of the AF from Example 1.

We provide a full analysis of the computational complexity of these generalized existence problems.

This article is structured as follows. In Section II, we provide a description of the formal models of abstract AFs and its extension to incomplete AFs, and a formal definition of the nonemptiness problem. This is followed by a full analysis of the computational complexity of its generalizations for incomplete AFs in Section III. In Section IV, we summarize our results and point out some related work in this field.

MODEL

In this section, we describe the notion of (abstract) AF formally, which was first presented by Dung.[1] Then, we extend this model to *incomplete* AFs, as has been done by Baumeister *et al.* [2]

Abstract AFs

An AF is a pair $\langle \mathcal{A}, \mathcal{R} \rangle$, where $\mathcal{A}$ is a finite set of arguments and $\mathcal{R}$ is a set of attacks between arguments, with $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. An argument a is said to *attack* an argument b if $(a, b) \in \mathcal{R}$. We call an argument a *acceptable with respect to a set* $S \subseteq \mathcal{A}$ if for each argument $b \in \mathcal{A}$ attacking a (i.e., $(b, a) \in \mathcal{R}$), there is an argument $c \in S$ that attacks b (i.e., $(c, b) \in \mathcal{R}$); we then say that a is *defended by* c *against its attacker* b.

An AF $\langle \mathcal{A}, \mathcal{R} \rangle$ can be illustrated by a directed graph with vertex set $\mathcal{A}$ and edge set $\mathcal{R}$.

Example 1. Let $AF = \langle \mathcal{A}, \mathcal{R} \rangle$ with $\mathcal{A} = \{a, b, c, d, e\}$ and $\mathcal{R} = \{(a, b), (a, d), (b, e), (c, a), (c, c), (d, c), (d, e), (e, d)\}$ be an AF. The corresponding graph is shown in Figure 1.

In addition to AFs, Dung[1] also introduced the notion of *semantics*—properties that a set of arguments should satisfy to be considered acceptable in an argumentation. A set of arguments that satisfies a given semantics is called an *extension*.

The first semantics we consider is CF. The idea behind it is that a set of acceptable arguments must not contain arguments that contradict (i.e., that attack) each other.

Definition 2. A set $S \subseteq \mathcal{A}$ is *conflict free* (CF) if there are no arguments a and b in S such that $(a, b) \in \mathcal{R}$.

For every AF, there is at least one CF set. In particular, the empty set is always CF.

CF sets that defend all their elements against all attackers are called *admissible*—commonly, a set of arguments is deemed unacceptable if it is not at least admissible.

Definition 3. We call a CF set $S \subseteq \mathcal{A}$ *admissible* (AD) if every argument $a \in S$ is acceptable with respect to S.

AD sets that are closed with respect to defense are called *complete* extensions of the AF.

Definition 4. An AD set $S \subseteq \mathcal{A}$ is *complete* (CP) if it contains every argument that is acceptable with respect to S.

The unique set of arguments that is minimal (with respect to set inclusion) among all complete extensions is the *grounded* extension. It can be deterministically constructed using an AF's characteristic function.

Definition 5. The *characteristic function* F_{AF} of an AF $AF = \langle \mathcal{A}, \mathcal{R} \rangle$ is a function $F_{AF} : 2^{\mathcal{A}} \to 2^{\mathcal{A}}$ defined by

$$F_{AF}(S) = \{a \in \mathcal{A} \mid a \text{ is acceptable with respect to } S\}.$$

A set $S \subseteq \mathcal{A}$ is *grounded* (GR) if S is the least fixed point of the characteristic function of AF.

On the other hand, the maximal AD (or, equivalently, maximal complete) sets of arguments are called *preferred* extensions.

Definition 6. A set $S \subseteq \mathcal{A}$ is called *preferred* (PR) if it is a maximal AD set.

The *stable* extensions are the conflict-free sets that attack all its nonmembers.

Definition 7. We call a CF set $S \subseteq \mathcal{A}$ *stable* (ST) if for each $b \in \mathcal{A} \setminus S$, there is at least one argument $a \in S$ with $(a, b) \in \mathcal{R}$.

Unfortunately, a stable extension may not always exists, which is why Caminada[12] introduced the *semi-stable* semantics as a less restrictive alternative. In order to define it, we will first introduce the notion of *range*.

Definition 8. We define the *range* of a set $S \subseteq \mathcal{A}$ as $S \cup S^{+}$, with $S^{+} = \{b \in \mathcal{A} \mid \exists a \in S : (a, b) \in \mathcal{R}\}$.

The *semistable* extensions can now be defined as the AD sets that maximize their range.

Definition 9. The *semistable* sets $S \subseteq \mathcal{A}$ (SST) are the AD sets for which $S \cup S^+$ is maximal (w.r.t. set inclusion).

For comparison, the stable extensions can be characterized as the AD sets S where $S \cup S^+ = \mathcal{A}$.

A concept related to semistable semantics is the notion of *stage* semantics, introduced by Verheij.[13]

Definition 10. The *stage sets* $S \subseteq \mathcal{A}$ (STG) are the conflict-free sets for which $S \cup S^+$ is maximal (w.r.t. set inclusion).

Now, let us summarize the relations among the various semantics defined previously. The complete, grounded, stable, semistable, and preferred semantics also satisfy CF and admissibility. Every preferred, grounded, stable, or semistable extension is also complete, and every stable and semistable extension is preferred. A stable extension is also semistable and a stage extension, and if a stable extensions exists, the semistable and stage extensions coincide with the stable ones.

For every AF, there always exists a conflict-free, admissible, preferred, complete, semistable, stage, and grounded extension—however, this may be the empty set. For the stable semantics, existence is not guaranteed. Also, while the grounded extension is unique for every AF, for each of the other semantics there may be more than one extension in a given AF.

Example 11. We continue Example 1. The CF sets of our example AF are $\{a, e\}$, $\{b, d\}$, and all their subsets. This AF has no nonempty AD set, since no nonempty, CF set of arguments can defend all its members: $\{a\}$ and $\{a, e\}$ lack an attack against c; $\{e\}$ lacks an attack against b; and $\{b\}$, $\{d\}$, and $\{b, d\}$ lack an attack against a. Therefore, the AF also has no nonempty extension for either of the complete, grounded, preferred, stable, or semistable semantics. However, $\{a, e\}$ and $\{b, d\}$ are both stage extensions.

We will study the *existence* problem restricted to nonempty extensions (i.e., the *nonemptiness* problem) for the abovementioned semantics in incomplete AFs. For standard AFs (in which there is no uncertainty regarding the arguments or attacks), the existence problem was first discussed by Dimopoulos and Torres[14] and later on by Dunne and Wooldridge.[15] Let $s \in \{\mathrm{CF}, \mathrm{AD}, \mathrm{CP}, \mathrm{GR}, \mathrm{PR}, \mathrm{ST}, \mathrm{SST}, \mathrm{STG}\}$ be any of the above semantics and define the following problem:

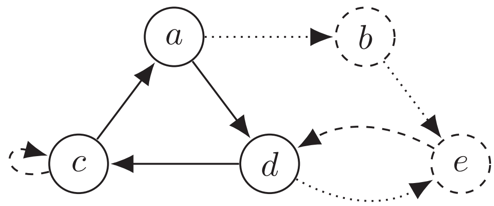

FIGURE 2. Graph representation of the IAF from Example 13, where uncertain arguments are drawn as dashed nodes, uncertain attacks are dashed arrows, and conditionally definite attacks are dotted arrows.

s-Nonemptiness (s-NE)	
Given:	An argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$.
Question:	Does there exist a nonempty set $S \subseteq \mathcal{A}$ satisfying the conditions specified by semantics s?

Incomplete AFs

We now turn to the model of *incomplete* AF that, based on earlier work by Coste-Marquis *et al.*[3] is due to Baumeister *et al.*.[2]

Definition 12. An *incomplete* AF (IAF) $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ partitions both the argument set and the attack set into *definite* ($\mathcal{A}$ and $\mathcal{R}$) and *uncertain* ($\mathcal{A}^?$ and $\mathcal{R}^?$) elements, where $\mathcal{R} \subseteq (\mathcal{A} \cup \mathcal{A}^?) \times (\mathcal{A} \cup \mathcal{A}^?)$ and $\mathcal{R}^? \subseteq (\mathcal{A} \cup \mathcal{A}^?) \times (\mathcal{A} \cup \mathcal{A}^?)$ are disjoint attack relations.

The uncertain elements of an IAF can be added to or left out of an AF. This operation is formally captured by *completions* $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ of an incomplete AF $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$, which must satisfy $\mathcal{A} \subseteq \mathcal{A}^* \subseteq \mathcal{A} \cup \mathcal{A}^?$ and $\mathcal{R}|_{\mathcal{A}^*} \subseteq \mathcal{R}^* \subseteq (\mathcal{R} \cup \mathcal{R}^?)|_{\mathcal{A}^*}$. We call $\mathcal{R}'|_{\mathcal{A}'}$ a *restriction of an attack relation* $\mathcal{R}'$ *to a set* $\mathcal{A}'$ *of arguments*, with $\mathcal{R}'|_{\mathcal{A}'} = \{(a, b) \in \mathcal{R}' \mid a, b \in \mathcal{A}'\}$. If $\mathcal{R}^? = \emptyset$, an IAF is said to be a purely *attack* IAF and we omit the empty uncertain attack relation in $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ and simply write $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R} \rangle$ instead. Similarly, if $\mathcal{A}^? = \emptyset$, it is said to be a purely *attack* IAF and we omit the empty uncertain argument set in $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ and simply write $\langle \mathcal{A}, \mathcal{R}, \mathcal{R}^? \rangle$ instead.

Example 13. Let IAF $= \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ be an IAF that has the argument sets $\mathcal{A} = \{a, c, d\}$ and $\mathcal{A}^? = \{b, e\}$ and the attack relations $\mathcal{R} = \{(a, b), (a, d), (b, e), (c, a), (d, c), (d, e)\}$ and $\mathcal{R}^? = \{(c, c), (e, d)\}$. The corresponding graph is shown in Figure 2. Following Baumeister *et al.* [2] we draw uncertain arguments and attacks as dashed nodes and edges, respectively. The dotted attacks (a, b), (b, e), and (d, e) are elements of

$\mathcal{R}$, but incident to at least one uncertain argument. These attacks are definite, provided that both incident arguments occur in the completion—that is, we are forced to add these attacks to a completion whenever both incident arguments are present in it. On the other hand, if an uncertain argument is not in the completion, we cannot include any incident attack in that completion. Such attacks are called *conditionally definite* by Baumeister *et al.* [2]

When including all uncertain elements of IAF, we obtain as a completion the AF of Example 1. Thus, it is clear that not every completion of IAF has a nonempty extension for any of the AD, complete, grounded, stable, or preferred semantics. On the other hand, in the completion that includes the argument e and the attack (e, d), but excludes the argument b and the attack (c, c), the set $\{c, e\}$ is stable (and, therefore, also semistable, stage, preferred, complete, and admissible), so IAF does indeed have completions where a nonempty extension for these semantics exists. However, there is no completion of IAF that allows for a nonempty grounded extension, since that would require at least one unattacked argument, but every argument in our example is attacked by at least one argument in every completion.

POSSIBLE AND NECESSARY EXISTENCE IN INCOMPLETE AFS

In this section, we will formally define the problems of possible nonemptiness and necessary nonemptiness and study them in terms of their complexity.

Problem Definitions and Some Prerequisites

The semantics for IAFs are defined on completions. We use the idea of *possibly* and *necessarily* satisfied properties. A property is *possibly* satisfied if there is at least one completion of the IAF in which this property is satisfied. A property is *necessarily* satisfied if it is satisfied in every completion of the IAF. Following this scheme, we can introduce a "possible" and a "necessary" variant of the nonempty existence problem for IAFs for each semantics s defined in Section II-A. We start with the *possible* variant:

s-Possible-Nonemptiness (s-PosNe)	
Given:	An incomplete AF $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$.
Question:	Does a completion $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ of IAF exist such that there exists a nonempty s extension in AF^*?

For the *necessary* variant of this problem, we ask if for *all* completions the conditions of the s-Nonemptiness problem are satisfied:

s-Necessary-Nonemptiness (s-NecNe)	
Given:	An incomplete AF $IAF = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$.
Question:	For all completions $AF^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle$ of IAF, is there a nonempty s extension in AF^*?

Our goal is to study these problems in terms of their computational complexity, as has been done by Baumeister *et al.*[2] for the possible and necessary verification problems and by Baumeister *et al.*[7] for the possible and necessary credulous and skeptical acceptance problems. The relevant complexity classes all belong to the polynomial hierarchy introduced by Meyer and Stockmeyer[16,17] whose zeroth level is P, whose first level consists of NP and coNP, and whose second level consists of $\Sigma_2^p = \mathrm{NP}^{\mathrm{NP}}$ and $\Pi_2^p = \mathrm{coNP}^{\mathrm{NP}}$. We assume the reader to be familiar with the concepts of hardness and completeness based on polynomial-time many-one reducibility, and we refer the reader to, e.g., the textbooks by Papadimitriou[18] and Rothe[19] for more background on computational complexity. We will use a restricted version of the quantified satisfiability (QSAT) problem for our reductions showing Π_2^p-hardness. Specifically, employing and extending a general construction based on the work of Dimopoulos and Torres[14] and Baumeister *et al.*,[7] we will use Π_2SAT, the canonical problem for the complexity class Π_2^p, which is defined as follows.

Π_2SAT	
Given:	A 3-CNF formula φ over a set $X \uplus Y$ of propositional variables.
Question:	For all assignments τ_X on X, is there an assignment τ_Y on Y such that $\varphi[\tau_X, \tau_Y] = \mathtt{true}$?

In this problem definition, X and Y are disjoint sets of propositional variables, φ is a formula in 3-CNF (conjunctive normal form with at most three literals per clause) over these variables, τ_S is a truth assignment on a set of literals S associated with the variables in X and Y (i.e., a mapping $\tau_S : S \to \{\mathtt{true}, \mathtt{false}\}$), and $\varphi[\tau_S]$ is the truth value that φ evaluates to under the assignment τ_S. Restricting Π_2SAT by fixing $X = \emptyset$ in its definition provides the NP-complete problem 3-SAT, which we will use for showing NP-hardness.

Analogously to the construction of Baumeister *et al.*,[7] we translate a QSAT instance to one of two versions of IAFs in Definition 14, the first being purely argument-incomplete (which, recall, means $\mathcal{R}^? = \emptyset$) and the other purely attack-incomplete (which, recall, means $\mathcal{A}^? = \emptyset$). We use this construction later in our proofs establishing the complexity of the problems studied in this section.

Definition 14. Given a Π_2SAT instance (φ, X, Y) with $\varphi = \bigwedge_i c_i$ and $c_i = \bigvee_j \alpha_{i,j}$ for each clause c_i, where $\alpha_{i,j}$ are the literals in clause c_i, an argument IAF $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}\rangle$ representing (φ, X, Y) is defined by

$$\begin{aligned}
\mathcal{A} &= \{\bar{x}_j \mid x_j \in X\} \cup \{y_k, \bar{y}_k \mid y_k \in Y\} \\
&\quad \cup \{c_i \mid c_i \text{ in } \varphi\} \cup \{\varphi, d, g\} \\
\mathcal{A}^? &= \{x_j \mid x_j \in X\} \\
\mathcal{R} &= \{(x_j, \bar{x}_j) \mid x_j \in X\} \cup \{(y_k, \bar{y}_k), (\bar{y}_k, y_k) \mid y_k \in Y\} \\
&\quad \cup \{(x_j, c_i) \mid x_j \text{ in } c_i\} \cup \{(\bar{x}_j, c_i) \mid \neg x_j \text{ in } c_i\} \\
&\quad \cup \{(y_k, c_i) \mid y_k \text{ in } c_i\} \cup \{(\bar{y}_k, c_i) \mid \neg y_k \text{ in } c_i\} \\
&\quad \cup \{(c_i, \varphi) \mid c_i \in \varphi\} \cup \{(\varphi, d)\} \\
&\quad \cup \ \{(d, a) \mid a \in (\mathcal{A} \cup \mathcal{A}^?) \setminus \{\varphi\}\}.
\end{aligned}$$

Similarly, given (φ, X, Y) as abovementioned, an attack-incomplete AF $\langle \mathcal{A}, \mathcal{R}, \mathcal{R}^?\rangle$ representing (φ, X, Y) is defined by

$$\begin{aligned}
\mathcal{A} &= \{x_j, \bar{x}_j \mid x_j \in X\} \cup \{y_k, \bar{y}_k \mid y_k \in Y\} \\
&\quad \cup \{c_i \mid c_i \text{ in } \varphi\} \cup \{\varphi, d, g\} \\
\mathcal{R} &= \{(\bar{x}_j, x_j) \mid x_j \in X\} \cup \{(y_k, \bar{y}_k), (\bar{y}_k, y_k) \mid y_k \in Y\} \\
&\quad \cup \{(x_j, c_i) \mid x_j \text{ in } c_i\} \cup \{(\bar{x}_j, c_i) \mid \neg x_j \text{ in } c_i\} \\
&\quad \cup \{(y_k, c_i) \mid y_k \text{ in } c_i\} \cup \{(\bar{y}_k, c_i) \mid \neg y_k \text{ in } c_i\} \\
&\quad \cup \{(c_i, \varphi) \mid c_i \in \varphi\} \cup \{(\varphi, d)\} \\
&\quad \cup \{(d, a) \mid a \in \mathcal{A} \setminus \{\varphi\}\} \\
\mathcal{R}^? &= \{(g, \bar{x}_j) \mid x_j \in X\}.
\end{aligned}$$

Arguments c_i are called *clause arguments* and arguments x_j, $\bar{x}_j$, y_k, and $\bar{y}_k$ are called *literal arguments*. For every literal argument z_ℓ, there is a counterargument $\bar{z}_\ell$. For a given truth assignment τ_X on X, we obtain a completion AF^{τ_X} from the constructed IAFs abovementioned. For the argument-incomplete case, we construct $AF^{\tau_X} = \langle \mathcal{A}^{\tau_X}, \mathcal{R}^{\tau_X}\rangle$, with $x_j \in \mathcal{A}^{\tau_X} \Leftrightarrow \tau_X(x_j) = \texttt{true}$. For the attack-incomplete case, we generate $AF^{\tau_X} = \langle \mathcal{A}^{\tau_X}, \mathcal{R}^{\tau_X}\rangle$, with $\mathcal{A}^{\tau_X} = \mathcal{A}$ and $(g, \bar{x}_j) \in \mathcal{R}^{\tau_X} \Leftrightarrow \tau_X(x_j) = \texttt{true}$. For full assignments τ_X and τ_Y, we denote a corresponding set of arguments by $\mathcal{A}^{\tau_X}[\tau_X, \tau_Y] = \{x_j \mid \tau_X(x_j) = \texttt{true}\} \cup \{\bar{x}_j \mid \tau_X(x_j) = \texttt{false}\} \cup \{y_k \mid \tau_Y(y_k) = \texttt{true}\} \cup \{\bar{y}_k \mid \tau_Y(y_k) = \texttt{false}\}$ in the completion AF^{τ_X}.

Note that argument g is without effect in the argument-incomplete version of the translation and is only included there to avoid case distinctions between the two translations.

Lemma 15 now shows the central role that argument φ has in the construction for the existence of extensions.

Lemma 15. Let (φ, X, Y) be a Π_2SAT instance, and let $\langle \mathcal{A}, \mathcal{R}, \mathcal{R}^?\rangle$ or $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}\rangle$ be an incomplete AF created for the instance according to Definition 14. Let τ_X be an assignment on X. In the completion AF^{τ_X}, the argument φ has to be in every nonempty extension that satisfies the semantics $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{GR}, \mathrm{PR}, \mathrm{ST}, \mathrm{SST}\}$.

Proof. Argument d attacks every argument except φ, but d cannot occur in any extension because it attacks itself and, therefore, violates CF. So if there is a nonempty s extension, it needs to have a defending attack against d and φ is the only argument attacking d. □

Next, Lemma 16 provides a sufficient and necessary criterion for the existence of nonempty extensions that correspond to satisfying assignments of the Π_2SAT formula φ.

Lemma 16. Let (φ, X, Y) be a given Π_2SAT instance and let τ_X and τ_Y be assignments on X and Y, respectively. Let *IAF* be an incomplete *AF* created by Definition 14 for (φ, X, Y). Let AF^{τ_X} be its completion corresponding to τ_X and let $\mathcal{A}^{\tau_X}[\tau_X, \tau_Y]$ be the set of literal arguments corresponding to the total assignment.

1) If $\varphi[\tau_X, \tau_Y] = \texttt{true}$, then there exists a nonempty extension $\mathcal{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{g, \varphi\}$ in AF^{τ_X} which is AD, complete, preferred, semistable, and stable.
2) If $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for all assignments τ_Y on Y, then AF^{τ_X} has no nonempty AD set (and, therefore, no nonempty complete, preferred, semistable, or stable extension).

Proof. 1) Assume that $\varphi[\tau_X, \tau_Y] = \texttt{true}$. We show that $\mathcal{E} = \mathcal{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{g, \varphi\}$ is a stable extension in $\mathcal{A}^{\tau_X}$. This will imply that this extension is also AD, complete, preferred, and semistable.

Consider the argument-incomplete variant first, where the completion AF$^{\tau_X}$ includes an uncertain argument x_j if and only if $\tau_X(x_j) = \texttt{true}$. It is apparent that $\mathcal{E} = \mathcal{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{g, \varphi\}$ is CF, since there are no attacks among literal arguments of different literals g, or φ. It remains to show that all arguments in $\mathcal{A}^{\tau_X} \setminus \mathcal{E}$ are attacked by $\mathcal{E}$. First, d is attacked by $\varphi \in \mathcal{E}$. Every literal argument $\bar{x}_j$ with $\tau_X(x_j) = \texttt{true}$ is attacked by its counterpart $x_j \in \mathcal{E}$. Every x_j with $\tau_X(x_j) = \texttt{false}$ is not included in the completion. Every $\bar{y}_k$ with $\tau_Y(y_k) = \texttt{true}$ is attacked by its counterpart $y_k \in \mathcal{E}$, and every y_k with $\tau_Y(y_k) = \texttt{false}$ is attacked by its counterpart $\bar{y}_k \in \mathcal{E}$. Finally, since $\varphi[\tau_X, \tau_Y] = \texttt{true}$, every clause c_i in φ has at least one satisfied literal, whose corresponding literal argument is in $\mathcal{E}$ due to its construction. Thus, every clause argument c_i is attacked by that literal argument in $\mathcal{E}$, and we have shown that $\mathcal{E}$ is stable in $\mathcal{A}^{\tau_X}$.

For the attack-incomplete variant, the completion AF^{τ_X} includes an uncertain attack $(g, \bar{x}_j)$ if and only if $\tau_X(x_j) = \texttt{true}$. The only difference compared to the argument-incomplete variant is that the literal arguments x_j with $\tau_X(x_j) = \texttt{false}$ are included in the completion and need to be attacked by $\mathcal{E}$. This is the case, since $\bar{x}_j \in \mathcal{E}$ whenever $\tau_X(x_j) = \texttt{false}$.

2) Assume that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for all assignments τ_Y on Y. We show that AF^{τ_X} has no nonempty AD set. This will imply that there is also no nonempty complete, preferred, or stable extension.

Again, let us consider the argument-incomplete variant first. For contradiction, assume that $\mathcal{E}$ is a nonempty AD set in AF^{τ_X}. From Lemma 15, we know that φ must be in $\mathcal{E}$. Furthermore, φ must be defended by $\mathcal{E}$ against all attacks by the clause arguments c_i. The only arguments that can attack c_i are the literal arguments that correspond to literals in c_i. Let $X' \cup Y' \subseteq \mathcal{E}$ be a set of literal arguments that attacks all arguments c_i. This set induces assignments $\tau_{X'}$ on X' and $\tau_{Y'}$ on Y', respectively, such that $\tau_{X'}(x_j) = \texttt{true}$ if $x_j \in X'$, $\tau_{X'}(x_j) = \texttt{false}$ if $\bar{x}_j \in X'$, $\tau_{Y'}(y_k) = \texttt{true}$ if $y_k \in Y'$, and $\tau_{Y'}(y_k) = \texttt{false}$ if $\bar{y}_k \in Y'$. Recall that, for each literal argument corresponding to a literal in X, we have that, if $\tau_X(x_j) = \texttt{true}$, argument $\bar{x}_j$ is defeated by x_j without a chance of defense, so x_j could be a member of $\mathcal{E}$, but $\bar{x}_j$ cannot. If $\tau_X(x_j) = \texttt{false}$, argument x_j is excluded from the completion, so only $\bar{x}_j$ (and not x_j) could be a member of $\mathcal{E}$. In summary, we know that $\tau_{X'}$ must be a subassignment of τ_X. Since $X' \cup Y'$ attacks all clause arguments c_i, we know that assignment $\tau_{X'}$ can be extended to τ_X and $\tau_{Y'}$ can be extended to a full assignment τ_Y on Y, such that $\varphi[\tau_X, \tau_Y] = \texttt{true}$. However, this contradicts our assumption that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for all assignments τ_Y on Y.

The attack-incomplete variant uses uncertain attacks $(g, \bar{x}_j)$ instead of uncertain arguments x_j. Here, for each literal argument corresponding to a literal in X, we have that, if $\tau_X(x_j) = \texttt{true}$, argument $\bar{x}_j$ is defeated by g without a chance of defense, so x_j could be a member of AD sets, but $\bar{x}_j$ cannot. If $\tau_X(x_j) = \texttt{false}$, argument x_j is defeated by $\bar{x}_j$ without a chance of defense, so only $\bar{x}_j$ (and not x_j) could be a member of AD sets. Again, for a subset $X' \subseteq \mathcal{E}$ of literal arguments from X in an AD set $\mathcal{E}$, we know that $\tau_{X'}$ must be a subassignment of τ_X, and the remainder of the proof is analogous. □

With Lemmas 15 and 16, we see that the two constructions of Definition 14 behave similarly and can, therefore, be interchanged. For the sake of simplicity, we will only use the argument-incomplete case further on, which also yields the hardness of the problems for the general case with uncertainty for arguments *and* attacks.

Possible Nonempty Existence

Before presenting our hardness results for the possible nonemptiness problem, we start with an easy polynomial-time algorithm for CF and the closely related stage semantics. Note that Proposition 17 also contains a result on the necessary nonemptiness problem, s-NecNe. This problem is actually considered only in a later section. However, since this proof is closely related to the proof for s-PosNe, we provide both results together in this section.

Proposition 17. s-NecNe and s-PosNe are in P for $s \in \{\mathrm{CF}, \mathrm{STG}\}$.

Proof. A set with only one argument and no self-attack is always conflict-free and also nonempty. Thus, in an AF, we only need to find one argument without a self-attack to get a nonempty conflict-free extension in that AF. Conversely, if all arguments in an AF have a self-attack, then no nonempty set of arguments can be CF.

Using these considerations, CF-PosNe can be solved by checking whether all arguments—both definite and uncertain—in the given incomplete AF have (conditionally) definite self-attacks. If yes, the answer to CF-PosNe is "no," and vice versa. Similarly, IAF $\in$ CF-NecNe if and only if IAF has a definite argument with no definite or uncertain self-attack. For the stage semantics, it is clear that a nonempty stage set exists in an AF if and only if a nonempty CF set exists, so the problems coincide. □

Proposition 18. For $s \in \{\mathrm{AD}, \mathrm{ST}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}\}$, s-PosNe is NP-complete.

Proof. To prove NP membership, we look at the quantifier representation. For each $s \in \{\mathrm{AD}, \mathrm{ST}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}\}$, s-PosNe can be written as the set of all IAF such that

$$(\exists \text{ completion } \mathrm{AF}^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle \text{ of IAF})(\exists \mathcal{E} : \emptyset \subset \mathcal{E} \subseteq \mathcal{A}^*)\\ [\mathcal{E} \text{ satisfies } s \text{ in } \mathrm{AF}^*].$$

These two existential quantifiers, which both are polynomially length-bounded in the size of the encoding of the given IAF, can be merged into one existential polynomially length-bounded quantifier. For all but the preferred and semistable semantics, the inner predicate is in P, so the problem s-PosNe is in NP. For the preferred semantics, however, existence of a nonempty preferred extension is equivalent to existence of a nonempty AD set, so the upper bounds coincide. For the semistable semantics, we look at AD sets with a maximal range. If there exists a nonempty AD set then there also exists a nonempty semistable set, as the range of any nonempty set is greater than the range of the empty set.

NP-hardness can be deduced from the NP-hardness of the s-NE problem, which itself follows from the NP-hardness of the existence problem, which has been shown by Chvátal[20] for the stable semantics and by Dimopoulos and Torres[14] for the other semantics. □

On the other hand, for the grounded semantics, possible nonemptiness is easy to solve, (Although we focus on the complexity class P only, we mention in passing that GR-PosNE is even in L, the complexity class capturing logarithmic space[18,19], which is a.k.a. the class of problems that have efficient "parallel" algorithms. Intuitively, note that GR-PosNE can be characterized as follows: $(\mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^?) \in \mathrm{GR\text{-}PosNE}$ if and only if $(\exists a \in (\mathcal{A} \cup \mathcal{A}^?))(\forall b \in \mathcal{A})[(b, a) \notin \mathcal{R}]$. For comparison, GR-NE can be similarly characterized as follows: $(\mathcal{A}, \mathcal{R}) \in \mathrm{GR\text{-}NE}$ if and only if $(\exists a \in \mathcal{A})(\forall b \in \mathcal{A})[(b, a) \notin \mathcal{R}]$. Since GR-NE is even known to be in L (see, e.g., the book chapter by Dvořák and Dunne[6]), so is GR-PosNE.) though not in a trivial way.

Theorem 19. GR-PosNE is in P.

Proof. It is well known[21] that an AF has a nonempty grounded extension if and only if it has at least one unattacked argument—if the AF has an unattacked argument, then this argument is a member of the grounded extension, and if it does not, then no argument can be acceptable with respect to the empty set, so the empty set is the grounded extension. Therefore, the GR-PosNE problem is equivalent to asking whether an incomplete AF has a completion, which has an unattacked argument.

First, given an IAF $\mathrm{IAF} = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$, we search in the minimal completion $\mathrm{AF}^* = \langle \mathcal{A}, \mathcal{R}|_{\mathcal{A}} \rangle$ that only contains the definite parts of the IAF. If AF^* has an unattacked argument, the answer for the GR-PosNE instance is clearly "yes." Otherwise, for every uncertain argument $x \in \mathcal{A}^?$, let $\mathrm{AF}^*_x = \langle \mathcal{A} \cup \{x\}, \mathcal{R}|_{\mathcal{A}\cup\{x\}} \rangle$ be the completion that contains only x and all definite elements. If any of these completions AF^*_x has an unattacked argument, we can again answer "yes." If none of them have an unattacked argument, it is clear that no completion can have one, since adding several uncertain arguments simultaneously or adding additional attacks cannot produce unattacked arguments that were not unattacked before. Therefore, we can answer "no" in this case.

This algorithm requires to check at most $|\mathcal{A}^?| + 1$ completions for unattacked arguments, which can be done in polynomial time. □

Necessary Nonempty Existence

We now turn to the necessary nonemptiness problem (besides CF-NECNE and STG-NECNE, which have already been handled in Proposition 17) and show that, for many semantics, it is hard for the second level of the polynomial hierarchy. We start with the upper bounds on the complexity.

Proposition 20. For each $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}, \mathrm{ST}\}$, we have $s\text{-NECNE} \in \Pi_2^p$.

Proof. To show membership of s-NECNE in Π_2^p, we can again look at the quantifier representation. For $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}, \mathrm{ST}\}$, we can formulate the problem s-NECNE as the set of all IAF such that

$$(\forall \text{ completion } \mathrm{AF}^* = \langle \mathcal{A}^*, \mathcal{R}^* \rangle \text{ of } IAF)(\exists \mathcal{E} : \emptyset \subset \mathcal{E} \subseteq \mathcal{A}^*)[\mathcal{E} \text{ satisfies } s \text{ in } \mathrm{AF}^*].$$

For this representation, we again use quantifiers that are polynomially length-bounded in the size of the encoding of the given IAF. The existentially quantified inner part is the problem s-NONEMPTINESS—this problem is NP-complete for $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}, \mathrm{ST}\}$, as shown by Chvátal[20] and Dimopoulos and Torres.[14] Hence, we have an existential quantifier followed by a statement checkable in polynomial time. This expression is preceded by a universal quantifier over completions. Both quantifiers together yield that s-NECNE is a problem in Π_2^p. □

Next, we prove the corresponding lower bound, utilizing the preparations done in Lemma 16.

Theorem 21. s-NECNE is Π_2^p-complete for $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{PR}, \mathrm{SST}, \mathrm{ST}\}$.

Proof. The Π_2^p upper bound of s-NECNE follows immediately from Proposition 20.

To show Π_2^p-hardness of s-NECNE, we reduce from the problem Π_2SAT. Given a Π_2SAT instance (φ, X, Y), we create an argument IAF $\mathrm{IAF} = \langle \mathcal{A}, \mathcal{A}^?, \mathcal{R} \rangle$ according to Definition 14. Assume that for every assignment τ_X on X there exists an assignment τ_Y on Y such that $\varphi[\tau_X, \tau_Y]$ is $\mathtt{true}$. Then, we have a "yes"-instance of Π_2SAT and we need to show that there is a nonempty stable (and, therefore, also admissible, complete, preferred, and semistable) extension $\mathcal{E}$ in

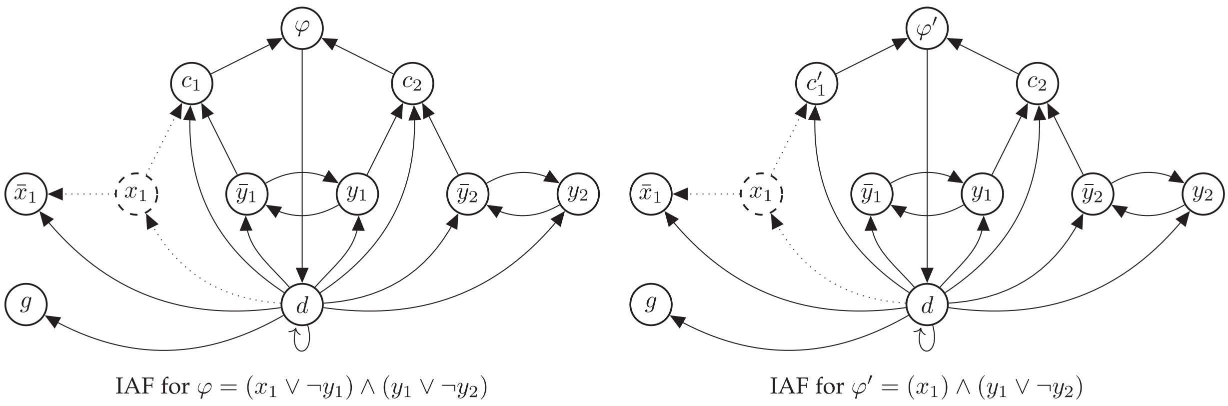

FIGURE 3. IAFs created for the Π_2SAT instances from Example 22 using the translation from Definition 14. The left IAF uses $\varphi = c_1 \wedge c_2 = (x_1 \vee \neg y_1) \wedge (y_1 \vee \neg y_2)$ and is a "yes"-instance of ST-NecNe, whereas the right IAF uses $\varphi' = c_1' \wedge c_2 = (x_1) \wedge (y_1 \vee \neg y_2)$ and is a "no"-instance of ST-NecNe.

every completion of IAF. This is provided by Lemma 16 where we have shown that if $\varphi[\tau_X, \tau_Y] = \texttt{true}$ then $\mathcal{E} = \mathcal{A}^{\tau_X}[\tau_X, \tau_Y] \cup \{g, \varphi\}$ is a nonempty stable extension in AF^{τ_X}. For the other direction of the proof, assume that we have a "no"-instance of Π_2SAT, i.e., there is an assignment τ_X on X such that for all assignments τ_Y on Y, $\varphi[\tau_X, \tau_Y]$ is $\texttt{false}$. Then, we need to show that there is a completion of IAF that has no nonempty AD set (and, therefore, also no nonempty complete, preferred, semistable, or stable extension), which again is provided by Lemma 16. □

Example 22. Consider the Π_2 SAT instance

$$\varphi = c_1 \wedge c_2 = (x_1 \vee \neg y_1) \wedge (y_1 \vee \neg y_2)$$

with $X = \{x_1\}$ and $Y = \{y_1, y_2\}$. This instance is a "yes"-instance of Π_2SAT, since for every assignment τ_X on X, there is an assignment τ_Y on Y such that $\varphi[\tau_X, \tau_Y] = \texttt{true}$—in particular, the assignment $\tau_Y(y_1) = \texttt{false}$ and $\tau_Y(y_2) = \texttt{false}$ satisfies φ irrespective of the assignment on X used.

Using our translation to IAFs, we get the representation $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R} \rangle$ with $\mathcal{A} = \{\bar{x}_1, y_1, \bar{y}_1, y_2, \bar{y}_2, c_1, c_2, \varphi, d, g\}$ and $\mathcal{A}^? = \{x_1\}$. Figure 3 shows the corresponding graph on the left. This ST-NecNe instance is a "yes"-instance, too: For τ_X^1 with $\tau_X^1(x_1) = \texttt{true}$, the set $\{\varphi, x_1, \bar{y}_1, \bar{y}_2, g\}$ is nonempty and stable in the completion $\mathrm{AF}^{\tau_X^1}$, and for τ_X^2 with $\tau_X^1(x_1) = \texttt{false}$, the set $\{\varphi, \bar{x}_1, \bar{y}_1, \bar{y}_2, g\}$ is nonempty and stable in the completion $\mathrm{AF}^{\tau_X^2}$, so there exists a nonempty stable extension in every completion.

If we modify the Π_2SAT instance to

$$\varphi' = c_1' \wedge c_2 = (x_1) \wedge (y_1 \vee \neg y_2)$$

(here, "$\vee \neg y_1$" is removed from the first clause), we get a "no"-instance of Π_2SAT, since for the assignment τ_X on X with $\tau_X(x_1) = \texttt{false}$, it holds that $\varphi[\tau_X, \tau_Y] = \texttt{false}$ for every assignment τ_Y on Y. The corresponding IAF is displayed in Figure 3 on the right. Here, the attack from argument $\bar{y}_1$ to c_1' is missing, and so the completion that omits argument x_1 has no nonempty stable extension, since no literal argument can defend φ' against the clause argument c_1' (recall that d cannot be in any stable extension due to its self-attack). Hence, this is a "no"-instance of ST-NecNe.

Finally, we turn to necessary nonemptiness for the grounded semantics, which is efficiently solvable.(Again, even though we focus on the complexity class P in Theorem 23 GR-NecNe in fact is in L. Specifically, referring to the case distinction in the upcoming proof of Theorem 23, note that our task in Case 1 can be done in L (just solve the GR-Ne instance $(\mathcal{A}, \mathcal{R} \cup \mathcal{R}^?)$, recalling that GR-Ne is in L); in Case 2 it is in L by solving the GR-Ne instance m times, $m \leq |\mathcal{A}^?|$; and in Case 3 it is in L because both previous cases are in L and we just apply one after the other).

Theorem 23. GR-NecNe is in P.

Proof. The idea behind this proof is to construct a critical completion that has a nonempty grounded extension only if *every* completion of that IAF has a nonempty grounded extension. If this critical

TABLE 1. Summary of complexity results.

s	s-NE		s-POSNE		s-NECNE	
CF	in P	*	in P	(Prop. 17)	in P	(Prop. 17)
STG	in P	6	in P	(Prop. 17)	in P	(Prop. 17)
GR	in P	21	in P	(Thm. 19)	in P	(Thm. 23)
AD	NP-c	14	NP-c	(Prop. 18)	Π_2^p-c	(Thm. 21)
CP	NP-c	14	NP-c	(Prop. 18)	Π_2^p-c	(Thm. 21)
PR	NP-c	14	NP-c	(Prop. 18)	Π_2^p-c	(Thm. 21)
SST	NP-c	6	NP-c	(Prop. 18)	Π_2^p-c	(Thm. 21)
ST	NP-c	20	NP-c	(Prop. 18)	Π_2^p-c	(Thm. 21)

For a complexity class $\mathcal{C}$, we use $\mathcal{C}$-c to denote "$\mathcal{C}$-complete." New results due to this paper are marked by the corresponding theorem, proposition, or corollary, and results due to previous work are marked by the respective reference. The result marked with * is straightforward and not formally proven.

completion has a nonempty grounded extension, this will allow a "yes" answer to the given GR-NECNE instance. Recall that, as already stated in the proof of Theorem 19, a nonempty grounded extension exists in a completion if and only if the completion has an unattacked argument.

Consider the following algorithm that defines a sequence AF_i of completions for a given instance $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle$ of GR-NECNE. AF_0 is the maximal completion that includes all uncertain elements of $\mathcal{A}^?$ and $\mathcal{R}^?$. Initially, let $i = 1$. Define $AF_i = AF_{(i-1)}$, but remove all arguments that were originally in $\mathcal{A}^?$ and that are unattacked in $AF_{(i-1)}$, and also remove attacks incident to these arguments. Repeat this step (with $i \leftarrow i+1$) until $AF_i = AF_{(i-1)}$ for some i. Then, $AF^* = AF_i$ is the desired critical completion.

If AF^* has no unattacked argument, then it also has no nonempty grounded extension and thus is a witness for $\langle \mathcal{A}, \mathcal{A}^?, \mathcal{R}, \mathcal{R}^? \rangle \notin$ GR-NECNE. If, on the other hand, AF^* has an unattacked argument x, then every completion must have an unattacked argument: in every completion, either x itself is unattacked, or if it isn't, then the completion must include some arguments from $\mathcal{A}^?$ that attack x. The only candidates are arguments that were removed by the algorithm. These can only be attacked by other arguments of the same kind—in particular, they are not attacked by definite arguments. Also, at least one of these arguments is not attacked by *any* other argument (which is removed in the first iteration of the algorithm). So, at least one of these arguments must be unattacked in the given completion. Note that excluding any additional uncertain attacks from completions can only make the existence of unattacked arguments in the completion more likely, so we do not need to consider this separately in the algorithm.

The algorithm iterates at most $|\mathcal{A}^?|$ times and performs only a polynomial number of steps per iteration, so GR-NECNE is in P $\square$

CONCLUSION

The main contribution of this work is to solve the open questions regarding the complexity of variants of the nonemptiness problem for IAFs. A summary of all results can be found in Table 1. When we compare the complexity of the incomplete problems with their corresponding complete versions, we can see that the complexity for the possible cases stays the same (due to collapsing existential quantifiers), whereas the complexity of the necessary cases is raised to Π_2^p for several semantics. The s-POSNE and s-NECNE problems for $s \in \{\mathrm{CF}, \mathrm{STG}, \mathrm{GR}\}$ have the same complexity as the respective base problem for standard AFs, while not relying on collapsing quantifiers. Interestingly, the Π_2^p-completeness result that we obtained for s-NECNE for the semantics $s \in \{\mathrm{AD}, \mathrm{CP}, \mathrm{PR}, \mathrm{ST}\}$ is the same as the complexity of necessary credulous acceptance for the same semantics[7] (the semistable semantics was not investigated by Baumeister *et al.*[7]). Necessary credulous acceptance for IAFs is the problem of deciding whether every completion of a given IAF has an s extension that contains a given target argument. This is a refinement of s-NECNE, which only requires the existence of a nonempty s extension in every completion, and does not demand membership of a given target argument. These coinciding complexities indicate that the hardness of necessary credulous acceptance in IAFs lies predominantly in making sure that an s extension exists in the first place, while the additional requirement of enforcing that the given target argument or set of arguments is contained in that extension does not add to the asymptotical complexity.

In addition to the original Dung[1] semantics, the semistable,[12], and the stage semantics[13] that we covered in this work, there are more semantics to study, such as the CF2 semantics[22] or the ideal semantics.[23] The complexity of generalized existence problems in IAFs for these semantics is an interesting task for future work. Cayrol *et al.*[24] introduced attack IAFs as *partial AFs* (PAFs); see also the forthcoming book chapter by Baumeister *et al.*[25] Instead of reducing the semantics of PAFs to the semantics of AFs via completions (like IAFs do), they define new semantics for partial AFs, which coincide with the completion-based approach for CF, but generally not for the more advanced semantics. Maher[26] presented a model of *strategic argumentation*, which simulates a game between a proponent P and an opponent O, where P tries to make an argument a accepted, while O tries the opposite. In this model, there are three sets of arguments, $\mathcal{A}_{Com}$ (common knowledge), $\mathcal{A}_P$ (arguments of P), and $\mathcal{A}_O$ (arguments of O), where $\mathcal{A}_P$ and $\mathcal{A}_O$ are comparable to uncertain arguments in IAFs. However, this model allows no uncertain attacks between arguments; every attack is definite. Furthermore, IAFs have some similarities to AF expansion due to Baumann and Brewka,[27], where new arguments and new attacks incident to at least one new argument may be added. Opposed to IAFs, the set of new arguments and attacks is not fixed, and new attacks among existing arguments are not allowed. Another variation of our model could be to add weights to the uncertain elements. So far, we tried to minimize extensions only with respect to the number of arguments in them. Also, in the model used here, arguments and attacks are treated essentially equally, which may not always be intended, especially not with uncertainty in the model. Previous work by Dunne *et al.*,[28] who looked at weighted attacks, could help to better refine our model. Similarly, Li *et al.*[29] followed a probabilistic approach, using probabilities both for arguments and attacks, and later on, Fazzinga *et al.*[30] studied the complexity of problems defined in this probabilistic model. Again, our model might benefit from refining it probabilistically in future work.

ACKNOWLEDGMENTS

This work was supported in part by Deutsche Forschungsgemeinschaft under Grants KE 1413/11-1, Grant RO 1202/14-2, and Grant RO 1202/21-1 and in part by the project "Online Participation" funded by the NRW Ministry for Innovation, Science, and Research. This article extends a preliminary version published at ECAI 2020. The authors would like to thank the reviewers for helpful comments.

REFERENCES

1. P. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and *n*-person games," *Artif. Intell.*, vol. 77, no. 2, pp. 321–357, 1995.
2. D. Baumeister, D. Neugebauer, J. Rothe, and H. Schadrack, "Verification in incomplete argumentation frameworks," *Artif. Intell.*, vol. 264, pp. 1–26, 2018.
3. S. Coste-Marquis, C. Devred, S. Konieczny, M. Lagasquie-Schiex, and P. Marquis, " On the merging of dung's argumentation systems," *Artif. Intell.*, vol. 171, no. 10, pp. 730–753, 2007.
4. D. Baumeister, J. Rothe, and H. Schadrack, " Verification in Argument-Incomplete Argumentation Frameworks," in *Proc. 4th Int. Conf. Algorithmic Decis. Theory*, Sep. 2015, pp. 359–376.
5. D. Baumeister, D. Neugebauer, and J. Rothe, " Verification in Attack-Incomplete Argumentation Frameworks," in *Proc. 4th Int. Conf. Algorithmic Decis. Theory*, Sep. 2015, pp. 341–358.
6. W. Dvořák and P. Dunne, "Computational Problems in Formal Argumentation and Their Complexity," in *Handbook Formal Argumentation*, P. Baroni, D. Gabbay, M. Giacomin, and L. van der Torre, Eds., vol. 1. Rickmansworth, U.K.: College Publications, 2018, pp. 631–687.
7. D. Baumeister, D. Neugebauer, and J. Rothe, "Credulous and skeptical acceptance in incomplete argumentation frameworks," in *Proc. 7th Int. Conf. Comput. Models Argument*, 2018, pp. 181–192.
8. Y. Dimopoulos, J. Mailly, and P. Moraitis, "Control argumentation frameworks," in *Proc. 32nd AAAI Conf. Artif. Intell.*, Feb. 2018, pp. 4678–4685.
9. A. Niskanen, D. Neugebauer, and M. Järvisalo, "Controllability of control argumentation frameworks," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 1855–1861.
10. B. Fazzinga, S. Flesca, and F. Furfaro, "Revisiting the notion of extension over incomplete abstract argumentation frameworks," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 1712–1718.
11. A. Niskanen, D. Neugebauer, M. Järvisalo, and J. Rothe, "Deciding acceptance in incomplete argumentation frameworks," in *Proc. 34th AAAI Conf. Artif. Intell.*, Feb. 2020, pp. 2942–2949.
12. M. Caminada, "Semi-stable semantics," in *Proc. 1st Int. Conf. Comput. Models Argument.*, Sep. 2006, pp. 121–130.

13. B. Verheij, "Two approaches to dialectical argumentation: Admissible sets and argumentation stages," in *Proc. 8th Dutch Conf. Artif. Intell.*, 1996, pp. 357–368.
14. Y. Dimopoulos and A. Torres, "Graph theoretical structures in logic programs and default theories," *Theor. Comput. Sci.*, vol. 170, no. 1, pp. 209–244, 1996.
15. P. Dunne and M. Wooldridge, " Complexity of Abstract Argumentation,", in *Argumentation in Artificial Intelligence* I. Rahwan and G. Simari, Eds. Berlin, Germany: Springer, 2009, ch. 5, pp. 85–104.
16. A. Meyer and L. Stockmeyer, "The equivalence problem for regular expressions with squaring requires exponential space," in *Proc. 13th IEEE Symp. Switching Automata Theory*, 1972, pp. 125–129.
17. L. Stockmeyer, "The polynomial-time hierarchy," *Theor. Comput. Sci.*, vol. 3, no. 1, pp. 1–22, 1976.
18. C. Papadimitriou, *Computational Complexity*, 2nd ed. Reading, MA, USA: Addison-Wesley, 1995.
19. J. Rothe, *Complexity Theory and Cryptology. An Introduction to Cryptocomplexity* (Ser. EATCS Texts in Theoretical Computer Science). New York, NY, USA: Springer-Verlag, 2005.
20. V. Chvátal, "On the computational complexity of finding a kernel,"Centre de Recherches Mathématiques, Université de Montréal, Montreal, QC, Canada, Tech. Rep. CRM-300, 1973.
21. S. Modgil and M. Caminada, " Proof theories and algorithms for abstract argumentation frameworks," in *Argumentation in Artificial Intelligence.* Berlin, Germany: Springer, 2009, pp. 105–129.
22. P. Baroni, M. Giacomin, and G. Guida, "SCC-recursiveness: A general schema for argumentation semantics," *Artif. Intell.*, vol. 168, no. 1/2, pp. 162–210, 2005.
23. P. Dung, P. Mancarella, and F. Toni, "A dialectic procedure for sceptical, assumption-based argumentation," in *Proc. 1st Int. Conf. Comput. Models Argument*, Sep. 2006, pp. 145–156.
24. C. Cayrol, C. Devred, and M. Lagasquie-Schiex, " Handling Ignorance in Argumentation: Semantics of Partial Argumentation Frameworks," in *Proc. 9th Eur. Conf. Symbolic Quantitative Approaches Reasoning Uncertainty*, Nov. 2007, pp. 259–270.
25. D. Baumeister, J. Rothe, and D. Neugebauer, "Collective acceptability in abstract argumentation," in *Handbook of Formal Argumentation*, D. Gabbay, M. Giacomin, G. Simari, and M. Thimm, Eds., vol. 2. Rickmansworth, U.K.: College Publications.
26. M. Maher, "Resistance to corruption of strategic argumentation," in *Proc. 30th AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 1030–1036.
27. R. Baumann and G. Brewka, "Expanding argumentation frameworks: Enforcing and monotonicity results," in *Proc. 3rd Int. Conf. Comput. Models Argument*, Sep. 2010, pp. 75–86.
28. P. Dunne, A. Hunter, P. McBurney, S. Parsons, and M. Wooldridge, "Weighted argument systems: Basic definitions, algorithms, and complexity results," *Artif. Intell.*, vol. 175, no. 2, pp. 457–486, 2011.
29. H. Li, N. Oren, and T. Norman, "Probabilistic argumentation frameworks," in *Proc. 1st Int. Workshop Theory Appl. Formal Argumentation*, Jul. 2011, pp. 1–16.
30. B. Fazzinga, S. Flesca, and F. Parisi, "On the complexity of probabilistic abstract argumentation," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Aug. 2013, pp. 898–904.

KENNETH SKIBA is currently a Research Assistant with Institute WeST (Web Science and Technologies), University of Koblenz-Landau, Koblenz, Germany. His research interests include formal argumentation theory, computational complexity theory, and conditional logic. He received the B.Sc. and M.Sc. degrees in computer science from Heinrich-Heine-Universität Düsseldorf, Düsseldorf, Germany, in 2017 and 2019, respectively. Contact him at kennethskiba@uni-koblenz.de.

DANIEL NEUGEBAUER is currently a Research Assistant with Heinrich-Heine-Universität Düsseldorf, Düsseldorf, Germany. His research interests include formal argumentation theory—with a focus on uncertainty, dynamics, and instantiation—as well as computational complexity theory and computational social choice. He received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Heinrich-Heine-Universität Düsseldorf, Germany, in 2012, 2014, and 2019, respectively. He is the corresponding author of this article. Contact him at daniel.neugebauer@hhu.de.

JÖRG ROTHE is currently a Professor and chair with the Department of Computer Science, Heinrich-Heine-Universität Düsseldorf, Düsseldorf, Germany. His research interests include computational complexity, computational social choice, argumentation theory, algorithmic game theory, fair division, and collective decision making. He received the Diploma degree in mathematics, and the Ph.D. and habilitation degrees both in computer science from Friedrich-Schiller-Universität Jena, Germany, in 1991, 1995, and 1999, respectively. Contact him at rothe@hhu.de.