Featured Argumentation Framework: Semantics and Complexity

Gianvincenzo Alfano, **Sergio Greco**, **Francesco Parisi** and **Irina Trubitsyna**
DIMES Department, University of Calabria, Rende, Italy
{g.alfano, greco, fparisi, i.trubitsyna}@dimes.unical.it

Abstract

Dung's Argumentation Framework (AF) has been extended in several directions to make knowledge representation and reasoning tasks more intuitive and/or expressive. We present a novel extension of AF called *Featured AF* (FAF), where each argument has associated a set of features expressed by means of unary and binary facts. In such a context, a query is expressed by means of a conjunctive relational calculus formula which is evaluated over the extensions of the FAF. Then, this framework is further expanded into the so-called *Extended FAF (EFAF)*, where a first-order logic formula (FOL) is used for reasoning over 'feasible' subframeworks that satisfy the FOL formula and *minimally* differ from the original framework. We investigate the computational complexity of verification and acceptance problems under several semantics and show that incomplete AF (iAF) frameworks, including *correlated iAF* and *constrained iAF*, are special cases of EFAF.

1 Introduction

Formal argumentation has emerged as one of the important fields in Artificial Intelligence [Bench-Capon and Dunne, 2007; Simari and Rahwan, 2009; Atkinson *et al.*, 2017]. Argumentation has applications in several contexts, e.g. modeling dialogues, negotiation [Dimopoulos *et al.*, 2019], persuasion [Prakken, 2009], and process mining [Fazzinga *et al.*, 2022b]. Dung's abstract Argumentation Framework (AF) is a simple yet powerful formalism for modeling disputes between two or more agents [Dung, 1995]. An AF consists of a set of *arguments* and a binary *attack* relation over the set of arguments that specifies the interactions between arguments: intuitively, if argument a attacks argument b, then b is acceptable only if a is not. Hence, arguments are abstract entities whose role is entirely determined by the attack relation. An AF can be seen as a directed graph, whose nodes represent arguments and edges represent attacks. Several argumentation semantics—e.g. *grounded* (gr), *complete* (co), *preferred* (pr), and *stable* (st) [Dung, 1995] —have been defined for AF, leading to the characterization of *σ-extensions*, that intuitively consist of the sets of arguments that can be collectively accepted under semantics $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$.

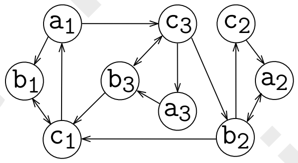

Figure 1: AF Λ of Example 1.

Various proposals have been made to extend the Dung's framework with the aim of better modeling the knowledge to be represented. The extensions include Bipolar AF [Nouioua and Risch, 2011; Nouioua, 2013; Villata *et al.*, 2012], AF with recursive attacks and supports [Cohen *et al.*, 2015; Gottifredi *et al.*, 2018; Cayrol *et al.*, 2018; Alfano *et al.*, 2024a], Dialectical framework [Brewka *et al.*, 2013], AF with preferences [Amgoud and Cayrol, 1998; Modgil and Prakken, 2013; Alfano *et al.*, 2023b] and constraints [Coste-Marquis *et al.*, 2006; Arieli, 2015; Alfano *et al.*, 2023c; Alfano *et al.*, 2024b], as well extensions for representing uncertain information. For the representation of uncertain information, two main extensions have been proposed: *incomplete AF* (iAF), where arguments and attacks may be uncertain [Baumeister *et al.*, 2018; Baumeister *et al.*, 2021], and *probabilistic AF* (PrAF), where arguments and attacks are associated with a probability [Dung and Thang, 2010; Li *et al.*, 2011; Hunter, 2012; Alfano *et al.*, 2023a; Fazzinga *et al.*, 2022a].

Our example concerns opinions about air pollution.

Example 1. Consider a scenario in which three air pollution experts, Alice (a), Bob (b), and Carl (c), are asked to propose short-, medium-, and long-term solutions to the problem. The experts have different general opinions: Alice supports industrial pollution regulation policies, Bob supports traffic limitation policies, and Carl promotes changes in human habits. Each expert expresses opinions for the short, medium, and long term, represented by the arguments x_1, x_2, x_3, respectively, where x denotes the first letter of the experts' name. These 9 arguments can be summarized as follows:

a_1: Impose immediate fines and sanctions on industries exceeding pollution limits;
a_2: Enforce stricter industrial regulations with compliance mechanisms;

a_3: Mandate the adoption of green technologies to achieve sustainable industrial practices;

b_1: Implement congestion charges and restrict access for high-polluting vehicles;

b_2: Expand public transportation to reduce private vehicle use;

b_3: Redesign cities to prioritize pedestrians, cyclists, and electric vehicles;

c_1: Launch awareness campaigns to encourage reduced car usage and energy conservation;

c_2 Integrate environmental education into schools and workplaces;

c_3: Promote societal norms through incentives for eco-friendly behavior and penalties for unsustainable actions.

The following attacks can be derived from considerations made by the three experts.

Short-term:	(a_1, b_1),(b_1, c_1),(c_1, a_1),(c_1, b_1);
Medium-term:	(a_2, b_2),(b_2, c_2),(c_2, a_2),(b_2, a_2);
Long-term:	(a_3, b_3),(b_3, c_3),(c_3, a_3),(c_3, b_3);
Cross-timeframes:	(a_1, c_3),(c_3, b_2),(b_2, c_1), (b_3, c_1).

The derived AF Λ (see Figure 1) has two stable (and preferred) extensions: E_1=$\{a_1, b_2, a_3\}$ and $E_2 = \{c_1, c_2, c_3\}$. □

The previous example shows how possible air pollution policies can be defined using AF under stable (and preferred) semantics. However, by modeling the problem by means of AF, we neglect important information that can be very useful. In our example, we disregard useful information about the text associated with the argument, the author (and her sex and age), and the type of policy (short-, medium-, long-term). Generally, an argument is associated with a sentence with a proper meaning. This means that, in adopting the abstraction of AF, a lot of relevant information, such as the argument's text, the author's id and gender, the date it was introduced, the topic (e.g. what the argument is about), the polarity (e.g. whether the sentence conveys a positive, negative or neutral sentiment), the polarity rating, the sentence style (e.g. veracity, sarcasm, irony), and many others, are lost.

As, in our opinion, these features should be made explicit and appropriately used, in this paper we propose an extension of AF called *Featured Argumentation Framework (FAF)*, where arguments might have associated features expressed by means of (first-order unary and binary) facts. Thus, the knowledge is represented through an underlying AF and a set of facts representing the features associated with arguments and with the AF's topology.

Example 2. Considering Example 1, each argument could have associated features such as *author* (e.g. *author*$(a_1$, *Alice*)), *text* (e.g. *text*$(c_2$, *"integrate environmental education into schools and workplaces"*)), supported *policy*, (e.g. *policy*$(b_1$, *traffic_limitation*)), *term* time (e.g. *term*$(c_3$, *long*)). □

In such a context, users pose queries consisting of conjunctive relational calculus formulae such as checking whether there exists an extension containing only arguments concerning policies proposed by the same expert (E_2 is an extension of such kind in Example 1). Herein, a query is evaluated with respect to the extensions of the underlying AF, each consisting of a subset of arguments with the associated features. As FAF may have multiple extensions, we study the credulous and skeptical acceptance of queries (a.k.a. brave and cautious acceptance, respectively, in the database (DB) context).

Consider now the case where there are two politicians, the former is interested only in short-term policies, the latter is interested only in policies regarding pollution regulations and traffic limitations. For reasoning about the first context (concerning short-term policies), we should focus on the subframework containing only the arguments a_1, b_1 and c_1, whereas for the second context we should consider the subframework containing only the arguments whose only policies are *pollution_regulation* and *traffic_limitation*. To this end, we propose an *Extended FAF (EFAF)*, where a First Order Logic (FOL) formula (also called constraint) is added to the FAF. The FOL formula is used to extract FAF *subframeworks* (subframeworks for short), that are subsets of the original FAF (i.e. the original EFAF without the constraint), that satisfy the constraints and differ from the original FAF by a minimal set or a minimum number of arguments and attacks.

Considering a multi-agent context, the FAF represents the general knowledge shared by all agents, whereas each constraint represent the knowledge of a given agent.

Example 3. The subframework containing only short-term policies can be defined by means of the FOL formula $\forall X.term(X, short)$, whereas the subframework considering only arguments denoting policies regarding both pollution regulation and traffic limitation can be defined by means of the FOL formula $\forall X.policy(X, pollution\_regulation) \vee policy(X,$ *traffic_limitation*$)$. On the other side, the formula $\forall X.policy(X, pollution\_regulation) \vee \forall X.policy(X,$ *traffic_limitation*$)$ allows determining two alternative subframeworks. □

The FOL formula can be viewed as an integrity constraint used to guarantee that available arguments and features correctly model the outside world. There is a tight connection with the Knowledge Base (KB) field as the facts denoting features and the AF topology represents the input database, C represents a set of constraints, and the subframeworks correspond to the subsets of the input facts that satisfy C and 'minimally' differ from the source database (called *repairs* in the KB context). Dealing with repairs in KBs is a critical area in knowledge management and AI [Arenas *et al.*, 1999; Greco *et al.*, 2001; Bertossi, 2011].

In our setting, a *subframework* satisfies the constraint and differs from the original FAF by a minimal set or a minimum number of arguments and attacks—these two minimality criteria are called *set* and *cardinality* (subframework) semantics, respectively. More specifically, a subframework is obtained by deleting a minimal set or a minimal number of arguments and attacks, so that the FOL formula is satisfied w.r.t the resulting FAF. Clearly, if deleting an argument then related attacks and features are deleted as well.

Therefore, an EFAF can be seen as representative of a set of FAFs obtained by extracting 'feasible' subframeworks. Then, its semantics can be defined in terms of the semantics of its subframeworks. This means that, similarly to other frameworks previously introduced in the literature (e.g. incomplete AF, probabilistic AF), the subframeworks can be seen as possible worlds. Thus, classical verification and acceptance problems (i.e. credulous and skeptical acceptance) need to be reconsidered in light of the presence of possible worlds, leading to the concepts of *possible* and *necessary verification*, as well as *possible* and *necessary* credulous/skeptical acceptance (cf. Definitions 7 and 8). Notice that, although the introduction of features allows knowing information about the content and meaning of arguments, their acceptance continues to be determined by the relationships between arguments, rather than the features' content.

Contributions. Our main contributions are as follows.

- We introduce an extension of AF called *Featured AF* (FAF) where each argument has a set of associated features defined through unary and binary facts. Queries over FAF are expressed by means of conjunctive relational calculus formulae which are evaluated over the FAF's extensions.
- We show that the complexity of acceptance problems in FAF does not increase with respect to that of AF.
- We further extend the FAF framework, leading to the so-called *Extended FAF* (EFAF), by adding *i*) features describing the topology of the underlying AF, and *ii*) a FOL formula which is used to select subframeworks (i.e. FAFs) satisfying the formula and differing from the original framework by a minimal set (or a minimal number) of elements (i.e. arguments and attacks).
- We introduce the verification and acceptance problems for EFAF and thoroughly investigate the complexity of these problems. A summary of our results for the verification and possible/necessary credulous/skeptical acceptance under argumentation semantics $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{pr}, \mathtt{st}\}$ and under set and cardinality subframework semantics are reported in Table 1 and Table 2, respectively. It turns out that standard reasoning problems in EFAF are strictly harder than the corresponding ones in incomplete AF [Baumeister *et al.*, 2021], correlated iAF [Fazzinga *et al.*, 2021a; Fazzinga *et al.*, 2021b] and constrained iAF [Mailly, 2024].
- Consistently with our complexity results, we show that incomplete AF, correlated and constrained iAF can be viewed as special cases of EFAF, in the sense that each of them can be rewritten into an extensions-equivalent EFAF, under both set and cardinality subframework semantics.

We assume the reader is familiar with the basic concepts of computational complexity.

2 Argumentation Framework

An abstract *Argumentation Framework* (AF) [Dung, 1995] is a pair $\langle A, R\rangle$, where A is a set of *arguments* and $R \subseteq A \times A$ is a set of *attacks*.

Given an AF $\Lambda = \langle A, R\rangle$ and a set $S \subseteq A$ of arguments, an argument $a \in A$ is said to be *i*) *defeated* w.r.t. S iff there exists $b \in S$ such that $(b, a) \in R$, and *ii*) *acceptable* w.r.t. S iff for every argument $b \in A$ with $(b, a) \in R$, there is $c \in S$ such that $(c, b) \in R$. The sets of arguments defeated and acceptable w.r.t. S are as follows (where Λ is fixed):

- $\mathit{Def}(S) = \{a \in A \mid \exists b \in S \,.\, (b, a) \in R\}$;
- $\mathit{Acc}(S) = \{a \in A \mid \forall b \in A \,.\, (b, a) \in R \Rightarrow b \in \mathit{Def}(S)\}$.

Given an AF $\langle A, R\rangle$, a set $S \subseteq A$ of arguments is said to be *conflict-free* iff $S \cap \mathit{Def}(S) = \emptyset$. Moreover, $S \subseteq A$ is said to be a *complete* (co) extension iff it is conflict-free and $S = \mathit{Acc}(S)$. A complete extension $S \subseteq A$ is said to be:

- *preferred* (pr) iff it is $\subseteq$-maximal;
- *stable* (st) iff it is a total preferred extension, i.e. a preferred extension such that $S \cup \mathit{Def}(S) = A$;
- *grounded* (gr) iff it is $\subseteq$-minimal.

In the following, if not specified otherwise, σ denotes any semantics in $\{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$. For any AF Λ and semantics σ, $\sigma(\Lambda)$ denotes the set of σ-extensions of Λ. All the above-mentioned semantics except the stable admit at least one extension (i.e. $\sigma(\Lambda) \neq \emptyset$), and the grounded admits exactly one extension (i.e. $|\mathtt{gr}(\Lambda)| = 1$) [Dung, 1995; Caminada, 2006]. For any AF Λ, it holds that $\mathtt{st}(\Lambda) \subseteq \mathtt{pr}(\Lambda) \subseteq \mathtt{co}(\Lambda)$, and $\mathtt{gr}(\Lambda) \subseteq \mathtt{co}(\Lambda)$.

For any AF $\Lambda = \langle A, R\rangle$ and set of arguments $S \subseteq A$, we use $\Lambda_{\downarrow S} = \langle A \cap S, R \cap (S \times S)\rangle$ to denote the AF obtained by projecting Λ over the arguments in S; we use $\sigma(\Lambda) \sqcap S = \{E \cap S \mid E \in \sigma(\Lambda)\}$ to denote the set of the intersections of the extensions in $\sigma(\Lambda)$ and S.

Given two AFs $\Lambda = \langle A, R\rangle$ and $\Lambda' = \langle A', R'\rangle$, we say that: (*i*) Λ' is included in Λ (denoted as $\Lambda' \sqsubseteq \Lambda$) iff $A' \subseteq A$ and $R' \subseteq R$, that is, Λ' can be obtained from Λ through the deletion of a (possibly empty) set of arguments and attacks; (*ii*) Λ' is strictly included in Λ (denoted as $\Lambda' \sqsubset \Lambda$) iff $\Lambda' \sqsubseteq \Lambda$ and $\Lambda' \neq \Lambda$, that is, Λ' can be obtained from Λ through the deletion of a non-empty set of arguments and attacks.

For any AF $\Lambda = \langle A, R\rangle$, semantics σ, and argument $a \in A$, we say that a is *credulously* (resp. *skeptically*) accepted (under semantics σ), denoted as $\mathit{CA}_\sigma(a, \Lambda)$ (resp. $\mathit{SA}_\sigma(a, \Lambda)$) if a belongs to at least one (resp. every) σ-extension of Λ.

We use CA_σ (resp. SA_σ), or CA (resp. SA) whenever σ is understood, to denote the credulous (resp. skeptical) acceptance problem, that is, the problem of deciding whether an argument is credulously (resp. skeptically) accepted. Clearly, for the grounded semantics, which has exactly one extension, these problems are identical (i.e. $\mathsf{CA}_{\mathtt{gr}} \equiv \mathsf{SA}_{\mathtt{gr}}$).

Example 4. Consider the AF $\Lambda = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{(\mathtt{a}, \mathtt{b}), (\mathtt{b}, \mathtt{a}), (\mathtt{a}, \mathtt{c}), (\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d}), (\mathtt{d}, \mathtt{c})\}\rangle$. Λ has 4 complete extensions: $E_0 = \emptyset$, $E_1 = \{\mathtt{a}, \mathtt{d}\}$, $E_2 = \{\mathtt{b}, \mathtt{d}\}$ and $E_3 = \{\mathtt{d}\}$. That is, $\mathtt{co}(\Lambda) = \{E_0, E_1, E_2, E_3\}$. E_0 is the grounded extension, E_1 and E_2 are stable (and preferred) extensions. Thus, a, b, d are credulously accepted for all semantics except the grounded; only d is skeptically accepted under stable and preferred semantics. □

3 Featured AF

We now introduce an extension of Dung's AF that consists of adding features associated with the arguments, leading to the concept of the Featured Argumentation Framework (FAF).

Syntax. We assume to have distinct countable sets of arguments $\mathbb{A}$, constants $\mathbb{D}$, variables $\mathbb{V}$, and (unary and binary) *base* predicates $\mathbb{P}_b$. Constants can be either natural numbers or alphanumeric strings starting with a lower-case letter, whereas variables are denoted by alphanumeric strings starting with an upper-case letter. We also assume to have the *predefined (or built-in)* unary predicate arg. $\mathbb{P}$ denotes the set of all predicates (base and built-in). Arguments, constants and variables are terms. An atom is of the form $p(a,b)$ (resp. $p(a)$), where p is a binary (resp. unary) predicate symbol and a,b are terms. Unary atoms take values from $\mathbb{A}$, (base) binary atoms take values from $\mathbb{A}\times\mathbb{D}$. A variable-free base atom is called a *fact*.

Given a set A of arguments, we use Arg_A to denote the set of built-in facts $\{arg(a) \mid a \in A\}$.

Definition 1. *A Featured AF (FAF) is a triple $\Omega = \langle A, R, F\rangle$ where $\langle A, R\rangle$ is an AF and F is a finite set of base facts.*

For an FAF $\langle A, R, F\rangle$, the set of base and built-in facts will be denoted by $F^A = F \cup Arg_A$.

The concept of inclusion defined earlier for AF can be immediately extended to FAF. Given two FAFs Ω and Ω', we write $\Omega' \sqsubseteq \Omega$ (resp. $\Omega' \sqsubset \Omega$) if Ω' can be obtained from Ω through the deletion of a possibly empty (resp. non-empty) set of arguments (as well as the related base facts) and attacks. That is, $\Omega' \sqsubseteq \Omega$ iff $A' \subseteq A$, $R' \subseteq R$, and $F' = F_{\downarrow_{A'}}$; moreover, $\Omega' \sqsubset \Omega$ iff $\Omega' \sqsubseteq \Omega$ and Ω' is not equal to Ω.

Semantics. Given a set of (base and built-in) facts $\mathcal{F}$ and a set of arguments S, we use $\mathcal{F}_{\downarrow_S} = \{p(a) \mid p(a) \in \mathcal{F} \wedge a \in S\} \cup \{p(a,b) \mid p(a,b) \in \mathcal{F} \wedge a \in S\}$ to denote the projection of $\mathcal{F}$ over S. That is, $\mathcal{F}_{\downarrow_S}$ contains only the facts in $\mathcal{F}$ referring to the arguments in S.

Definition 2 (Semantics)**.** *Given an FAF $\Omega = \langle A, R, F\rangle$ and a semantics σ, a σ-extension for Ω is any set $E \in \sigma(\langle A, R\rangle)$.*

As usual, the set of σ-extensions for Ω is denoted by $\sigma(\Omega)$. Thus, we have that $\sigma(\langle A, R, F\rangle) = \sigma(\langle A, R\rangle)$.

A query is a conjunctive relational calculus query defined over the database F^A, that is, it only uses the predicate symbols contained in F^A. A query is said to be boolean if all variables are (existentially) quantified. For instance, the boolean query $arg(a)$ checks the acceptance of argument a.

Definition 3 (Query acceptance)**.** *For any FAF $\Omega = \langle A, R, F\rangle$, semantics σ, and boolean query q, we say that q is (i) credulously accepted if there exists a set $E \in \sigma(\Omega)$ such that $F^A_{\downarrow E} \models q$,*[1] *and (ii) skeptically accepted if for every set $E \in \sigma(\Omega)$, $F^A_{\downarrow E} \models q$ holds.*

Example 5. Consider the FAF $\Omega = \langle A, R, F\rangle$ obtained from the AF $\langle A, R\rangle$ of Example 1, where F contains the following base facts $\{author(a_i, \mathit{Alice})$, $author(b_i, \mathit{Bob})$, $author(c_i, \mathit{Carl}) \mid i \in [1,3]\} \cup \{term(x_1, \mathit{short})$, $term(x_2, \mathit{medium})$, $term(x_3, \mathit{long}) \mid x \in \{a, b, c\}\}$. Assume we are interested to know whether there are stable extensions containing at least one short-term argument from Alice, thus we interested in the answer to the query $q = \exists X.author(X, \mathit{Alice}) \wedge\ term(X, \mathit{short})$. The answer to q is "yes" as there exists a stable-extension $E_1 = \{a_1, b_2, a_3\}$ for Ω and $F^A_{\downarrow E}$ contains the atoms $author(a_1, \mathit{Alice})$ and $term(a_1, \mathit{short})$. Although q is credulously accepted under stable semantics, it is not skeptically accepted, as there exists a stable-extension $E_2 = \{c_1, c_2, c_3\}$ for Ω s.t. $F^A_{\downarrow E'} \not\models q$. □

We use CA_σ (resp. SA_σ), to denote the credulous (resp. skeptical) acceptance problem in FAF under semantics σ, that is, the problem of deciding whether a boolean query q is credulously (resp. skeptically) accepted.

Proposition 1. *For FAF, it holds that:*

- *CA_σ is $i)$ in PTIME for $\sigma = \mathtt{gr}$, and $ii)$ NP-complete for $\sigma \in \{\mathtt{co}, \mathtt{st}, \mathtt{pr}\}$;*
- *SA_σ is $i)$ in PTIME for $\sigma \in \{\mathtt{gr}, \mathtt{co}\}$, $ii)$ coNP-complete for $\sigma = \mathtt{st}$, and $iii)$ Π_2^p-complete for $\sigma = \mathtt{pr}$.*

Thus, the complexity of acceptance problems in FAF and AF coincides.

4 Extended FAF

We now introduce the *Extended Featured Argumentation Framework (EFAF)*, that extends FAF with a FOL formula to be satisfied. The formula, defined on the argument features and the topology of the underlying AF, is used to define the subframeworks the user is interested in.

Syntax. Given an FAF $\Omega = \langle A, R, F\rangle$, we assume to also have the built-in binary predicate att, denoting attacks among arguments, and the *derived* binary predicates $path$, $\mathit{even\text{-}path}$ and $\mathit{odd\text{-}path}$ denoting paths over the argumentation graph $\langle A, R\rangle$; all these predicates take values from $A \times A$. Hereinafter, we use Att_R to denote the set $\{att(a,b) \mid (a,b) \in R\}$, and $\mathbb{P}$ to denote the set of all predicates (base, built-in, and derived ones). We use λ to denote the first-order language using base, built-in and derived predicates, and terms (i.e. arguments, constants and variables).

Definition 4. *An extended FAF (EFAF) is a tuple $\Delta = \langle A, R, F, C\rangle$ where $\langle A, R, F\rangle$ is an FAF and C is a first-order formula defined over λ.*

We use $F^{AR} = F \cup Arg_A \cup Att_R$ to denote the set containing base feature facts and built-in facts describing the topology of the underlying graph. Clearly, F^{AR} extends the set F^A defined earlier for FAF. In the following, for the sake of presentation, the formula C will be often represented as a set of formulae $\{c_1, ..., c_n\}$ corresponding to $c_1 \wedge \cdots \wedge c_n$.

Semantics. We start by defining the satisfaction of the formula C with respect to the underlying FAF $\langle A, R, F\rangle$. A base or built-in fact $p(a,b)$ (resp. $p(a)$) is satisfied w.r.t. a EFAF $\Delta = \langle A, R, F, C\rangle$ if it occurs in F^{AR}. Moreover, a derived fact $path(a,b)$ (resp. $\mathit{even\text{-}path}(a,b)$, $\mathit{odd\text{-}path}(a,b)$) is satisfied w.r.t. Δ if there exists a path (resp. even simple path, odd simple path) from a to b in the underlying AF $\langle A, R\rangle$. The role of the logical formula C in Δ is to represent a constraint (or a set of constraints) to be satisfied by F^{AR}. Intuitively, if the logical formula C is not satisfied, then the underlying FAF should be revised by computing subframeworks, that is, by (minimally) modifying the topology of the AF through the deletion of arguments and attacks, so that the formula is satisfied. Clearly, when deleting arguments, related features and attacks are deleted as well. In the following, we assume that the formula C is satisfiable, that is that there exists an FAF $\langle A, R, F\rangle$ such that $F^{AR} \models C$.

[1] Here $F^A_{\downarrow E}$ denotes the set $(F^A)_{\downarrow_E}$.

Definition 5 (Subframework)**.** *Given an EFAF $\Delta = \langle A, R, F, C\rangle$, a set-subframework (resp. cardinality-subframework) for Δ is an FAF $\langle A', R', F'\rangle \sqsubseteq \langle A, R, F\rangle$ such that, (i) $F^{AR}_{\downarrow A'} \models C$, and (ii) there is no FAF $\langle A'', R'', F''\rangle \sqsubseteq \langle A, R, F\rangle$ such that $F^{AR}_{\downarrow A''} \models C$ and $\langle A', R', F'\rangle \sqsubset \langle A'', R'', F''\rangle$ (resp. $|A'| + |R'| < |A''| + |R''|$).*

Thus, a set-subframework (s-subframework for short) is obtained by deleting a minimal set of arguments and attacks, whereas a cardinality subframework (c-subframework) is obtained by deleting a minimum number of arguments and attacks. Observe that, in computing subframeworks, feature facts are implicitly deleted through the deletion of the related arguments. This is line with repairing strategies exploited in databases, where the inconsistency of an attribute leads to the deletion of an entire tuple [Arenas *et al.*, 1999].

It is worth noting that the cardinality-based distance is able to discriminate between arguments and attacks, in the sense that the weight associated with an argument removal is also related to its degree (i.e., incoming/outgoing attacks): removing an argument with degree d has weight $d + 1$.

Example 6. Consider the EFAF $\Delta = \langle A, R, F, C\rangle$ obtained from the FAF $\langle A, R, F\rangle$ of Example 5, where the FOL formula $C = \forall X.term(X, \mathit{short})$, captures the subframeworks containing only short-term policies. The unique subframework (under set or cardinality semantics) is obtained by deleting every argument x such that $term(x, \mathit{short})$ is false, and its unique stable extension is $\{c_1\}$.

Assume now we want to fix exactly one term policy (either short, medium, or long), independently of the actual term's value. The FOL formula $C = \forall X, Y, Z, Z'\ term(X, Z) \wedge term(Y, Z') \Rightarrow Z = Z'$ allows to identify three s-subframeworks obtained by keeping only the arguments sharing the same term's value. □

The set of s-subframeworks (resp. c-subframeworks) for an EFAF Δ is denoted by $\mathcal{S}(\Delta)$ (resp. $\mathcal{C}(\Delta)$). The semantics of Δ is given by the set of its (FAF) subframeworks. We say that a subframework for Δ is obtained under set or cardinality semantics if it belongs to $\mathcal{S}(\Delta)$ or $\mathcal{C}(\Delta)$, respectively.

Notice that EFAF without base predicates is similar to correlated/constrained incomplete AF [Fazzinga *et al.*, 2021a; Fazzinga *et al.*, 2021b; Mailly, 2020], though the formulae defined in those works are propositional and allow only built-in predicates. However, the aim is different, as we are interested in dealing with subframeworks (which minimally differ from the original framework), whereas in the above-mentioned works all *completions* are first-class citizens.

5 Computational Complexity

We investigate the complexity of fundamental reasoning problems for EFAF. In particular, we study the verification, existence, and credulous/skeptical acceptance problems, that are usually considered for analyzing the complexity of argumentation frameworks.

We start with a lemma stating that the satisfaction of the logical formula C can be checked in PTIME in the size of the active domain consisting of constant values and arguments.

	AF	iAF		ciAF		EFAF			
σ	V_σ	PV_σ	NV_σ	PV_σ	NV_σ	$\mathsf{c\text{-}PV}_\sigma$	$\mathsf{s\text{-}PV}_\sigma$	$\mathsf{c\text{-}NV}_\sigma$	$\mathsf{s\text{-}NV}_\sigma$
$\mathtt{gr}$	P	P	P	NP-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
$\mathtt{co}$	P	P	P	NP-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
$\mathtt{st}$	P	P	P	NP-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
$\mathtt{pr}$	coNP-c	Σ_2^p-c	coNP-c	Σ_2^p-c	coNP-c	Θ_2^p-h,Σ_2^p	Σ_2^p-c	Θ_2^p-c	Π_2^p-c

Table 1: Complexity of the verification problems for AF, iAF, ciAF, and EFAF under semantics $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{pr}, \mathtt{st}\}$. For any complexity class C, C-c (resp. C-h) means C-complete (resp. C-hard). An interval C-h, C' means C-hard and in C'.

Notice that our results refer to data complexity, where a common assumption is that the number of variables in a formula is constant and thus the grounding is still in PTIME.

Lemma 1. *Given an EFAF $\Delta = \langle A, R, F, C\rangle$ and a set $A' \subseteq A$, checking whether $(F^{AR})_{\downarrow_{A'}} \models C$ can be done in PTIME.*

Observe that a given EFAF may admit zero, one, or multiple subframeworks. Subframework existence, verification, and acceptance problems can be defined analogously to those defined for iAF [Baumeister *et al.*, 2018; Baumeister *et al.*, 2021]. In fact, roughly speaking, the main differences with respect to iAF is that subframeworks correspond to completions which must satisfy a minimality criteria (either set or cardinality subframeworks semantics).

5.1 Existence Problems

We start by introducing the *existence problems* under set and cardinality subframeworks semantics. We use $\delta \in \{c, s\}$ to refer a subframeworks semantics: c standing for cardinality and s for set.

Definition 6. *Let $\delta \in \{c, s\}$, δ-EX is the problem of checking whether there exists a δ-subframework for a given EFAF.*

In our running example, it is easy to see that both problems are true. In general, deciding subframework existence is hard.

Theorem 1. $\mathsf{s\text{-}EX}$ *and* $\mathsf{c\text{-}EX}$ *are NP-complete.*

5.2 Verification Problems

We now characterize the complexity of the *verification problems* for EFAF, which are formally defined in what follows.

Definition 7. *Let $\Delta = \langle A, R, F, C\rangle$ be an EFAF, $S \subseteq A$ a set of arguments, $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{pr}, \mathtt{st}\}$ an argumentation semantics, and $\delta \in \{c, s\}$ a subframeworks semantics. Then:*

- *the δ-possible verification problem, denoted as δ-PV_σ, is the problem of checking whether S is a σ-extension in any δ-subframework for Δ;*
- *the δ-necessary verification problem, denoted as δ-NV_σ, is the problem of checking whether S is a σ-extension in all δ-subframeworks for Δ.*

Observe that for an EFAF of the form $\langle A, R, \emptyset, \emptyset\rangle$, that semantically coincides with the AF $\langle A, R\rangle$, $\mathsf{s\text{-}PV}_\sigma = \mathsf{c\text{-}PV}_\sigma = \mathsf{s\text{-}NV}_\sigma = \mathsf{c\text{-}NV}_\sigma$, which in turns coincide with the standard verification problem for AF, that is checking whether a given set S is a σ-extension of (the subframework) $\langle A, R\rangle$.

The following theorem states that the complexity of the verification problem for EFAF is generally harder than that for AF, whose complexity is summarized in the second column of Table 1 [Dvorak and Dunne, 2017]. Moreover, the results for the cardinality-based semantics are generally lower than those for the set-based semantics.

Theorem 2.

- c-PV$_\sigma$ *is* $i)$ Θ_2^p*-complete for* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}\}$*, and* $ii)$ Θ_2^p*-hard and in* Σ_2^p *for* $\sigma = \mathtt{pr}$*;*
- s-PV$_\sigma$ *is* Σ_2^p*-complete for* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- c-NV$_\sigma$ *is* Θ_2^p*-complete for* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- s-NV$_\sigma$ *is* Π_2^p*-complete for* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*.*

Notably, the verification problem for EFAF is computationally harder than that for *constrained iAF (ciAF)* [Mailly, 2020] (and thus iAF) under all the considered (argumentation and subframework) semantics, except for δ-PV$_{\mathtt{pr}}$ (with $\delta \in \{c, s\}$) for which the verification problems for EFAF and ciAF belong to the same class (Σ_2^p). On the other side, it can be shown that the complexity of verification for FAF is as that of AF, as any FAF coincides with its unique subframework and thus FAF verification collapses to AF verification.

5.3 Acceptance Problems

For AFs, the complexity of the credulous and skeptical acceptance problems has been investigated in [Dung, 1995] for the grounded semantics, in [Dimopoulos and Torres, 1996] for the stable semantics, and in [Dimopoulos and Torres, 1996; Dunne and Bench-Capon, 2002] for the preferred semantics. The complexity results for AFs are summarized in the second and third column of Table 2.

The following definition generalizes that of the credulous and skeptical acceptance problems for AF to the case of EFAF, analogously to what is done for (c)iAF.

Definition 8. *Let* $\Delta = \langle A, R, F, C\rangle$ *be an EFAF,* q *a boolean query,* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{pr}, \mathtt{st}\}$ *an argumentation semantics, and* $\delta \in \{c, s\}$ *a subframework semantics. Then:*

- *the* δ*-possible credulous acceptance problem, denoted as* δ-PCA$_\sigma$*, is the problem of checking whether* q *is credulously accepted in at least one* δ*-subframework for* Δ*;*
- *the* δ*-possible skeptical acceptance problem, denoted as* δ-PSA$_\sigma$*, is the problem of checking whether* q *is skeptically accepted in at least one* δ*-subframework for* Δ*;*
- *the* δ*-necessary credulous acceptance problem, denoted as* δ-NCA$_\sigma$*, is the problem of checking whether* q *is credulously accepted in all* δ*-subframeworks for* Δ*;*
- *the* δ*-necessary skeptical acceptance problem, denoted as* δ-NSA$_\sigma$*, is the problem of checking whether* q *is skeptically accepted in all* δ*-subframeworks for* Δ*.*

The next theorem provides a tight characterization of the complexity for all the above-defined acceptance problems.

Theorem 3.

- c-PCA$_\sigma$ *is* Θ_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- s-PCA$_\sigma$ *is* Σ_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- c-NCA$_\sigma$ *is* $i)$ Θ_2^p*-complete for* $\sigma = \mathtt{gr}$*, and* $ii)$ Π_2^p*-complete for any* $\sigma \in \{\mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- s-NCA$_\sigma$ *is* Π_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*;*
- c-PSA$_\sigma$ *is* $i)$ Θ_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}\}$*, and* $ii)$ Σ_2^p*-complete for any* $\sigma \in \{\mathtt{st}, \mathtt{pr}\}$*;*
- s-PSA$_\sigma$ *is* $i)$ Σ_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}\}$*, and* $ii)$ Σ_3^p*-complete for* $\sigma = \mathtt{pr}$*;*
- c-NSA$_\sigma$ *is* $i)$$\Theta_2^p$*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}\}$*, and* $ii)$ Π_2^p*-complete for* $\sigma = \mathtt{pr}$*.*
- s-NSA$_\sigma$ *is* Π_2^p*-complete for any* $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$*.*

In brief, except for the 7 cases highlighted with an asterisk (*) in Table 2 for which the complexity of EFAF is the same as that of (c)iAF, for all the other 25 cases considered the complexity of EFAF is higher than that of (c)iAF, even under cardinality subframework semantics whose complexity is systematically lower than or equal to that of the set semantics.

6 EFAF versus iAF-based Frameworks

As observed earlier, there are some connections between EFAF and iAF [Baumeister *et al.*, 2021]. Acceptance problems are defined similarly, in the sense that the semantics of iAF is defined by the set of completions, whereas that of EFAF is defined by the set of subframeworks (that can be viewed as a subset of completions by assuming arguments/attacks of EFAF to be uncertain). Moreover, EFAF without base features is similar to correlated/constrained iAF [Fazzinga *et al.*, 2021a; Fazzinga *et al.*, 2021b; Mailly, 2020], though the formulae defined in the latter frameworks are propositional and allow only built-in predicates. We investigate the relationship between EFAF and those frameworks, showing that iAF, as well as constrained and correlated iAF, are special cases of EFAF.

6.1 Incomplete Argumentation Framework

An *incomplete (abstract) Argumentation Framework* (iAF) is a tuple $\Psi = \langle A, B, R, T\rangle$, where A and B are disjoint sets of arguments, and R and T are disjoint sets of attacks between arguments in $A \cup B$. Arguments in A and attacks in R are said to be *certain*, while arguments in B and attacks in T are said to be *uncertain* [Baumeister *et al.*, 2018]. An iAF compa-ctly represents alternative AF scenarios, called *completions*.

Definition 9 (Completion)**.** A *completion* for an iAF $\Psi = \langle A, B, R, T\rangle$ is an AF $\Lambda = \langle A', R'\rangle$ where $A \subseteq A' \subseteq A \cup B$ and $R \cap (A' \times A') \subseteq R' \subseteq (R \cup T) \cap (A' \times A')$.

Observe that there may be completions that do not contain the attacks in R involving uncertain arguments, since those arguments do not belong to the considered completion.

The set of completions of iAF Ψ is denoted by $comp(\Psi)$.

Verification and acceptance problems, originally introduced in [Baumeister *et al.*, 2018; Baumeister *et al.*, 2021], are defined analogously to the ones of EFAF, with the only difference that instead of considering subframeworks and queries, completions and arguments are taken into account.

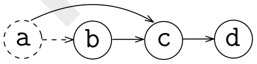

Figure 2: iAF of Example 7 (dashed elements are uncertain).

Example 7. Consider the iAF $\Psi = \langle\{\mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{\mathtt{a}\}, \{(\mathtt{a}, \mathtt{c}), (\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}, \{(\mathtt{a}, \mathtt{b})\}\rangle$ shown in Figure 2, where arguments $\mathtt{b}, \mathtt{c}, \mathtt{d}$ and attacks $(\mathtt{a}, \mathtt{c})$, $(\mathtt{b}, \mathtt{c})$ and $(\mathtt{c}, \mathtt{d})$ are certain, whereas argument $\mathtt{a}$ and attack $(\mathtt{a}, \mathtt{b})$ are uncertain.
Ψ has 3 completions, that are:

	AF/FAF		iAF/ciAF				EFAF							
σ	CA$_\sigma$	SA$_\sigma$	PCA$_\sigma$	NCA$_\sigma$	PSA$_\sigma$	NSA$_\sigma$	c-PCA$_\sigma$	s-PCA$_\sigma$	c-NCA$_\sigma$	s-NCA$_\sigma$	c-PSA$_\sigma$	s-PSA$_\sigma$	cNSA$_\sigma$	sNSA$_\sigma$
gr	P	P	NP-c	coNP-c	NP-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
co	NP-c	P	NP-c	Π_2^p-c	NP-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Π_2^p-c*	Π_2^p-c	Θ_2^p-c	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
st	NP-c	coNP-c	NP-c	Π_2^p-c	Σ_2^p-c	coNP-c	Θ_2^p-c	Σ_2^p-c	Π_2^p-c*	Π_2^p-c	Σ_2^p-c*	Σ_2^p-c	Θ_2^p-c	Π_2^p-c
pr	NP-c	Π_2^p-c	NP-c	Π_2^p-c	Σ_3^p-c	Π_2^p-c	Θ_2^p-c	Σ_2^p-c	Π_2^p-c*	Π_2^p-c	Σ_3^p-c*	Σ_3^p-c*	Π_2^p-c*	Π_2^p-c

Table 2: Complexity of possible and necessary credulous/skeptical acceptance under semantics $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{pr}, \mathtt{st}\}$ for AF, FAF, iAF, ciAF, and EFAF. For any complexity class $\mathcal{C}$, $\mathcal{C}$-c means $\mathcal{C}$-complete. All results for EFAF (and FAF) are new. The results with asterisk (*) refer to cases where the complexity of EFAF is the same as that of (c)iAF; for the other cases, the complexity of EFAF is higher than that of (c)iAF.

- $\Lambda_1 = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{(\mathtt{a}, \mathtt{b}), (\mathtt{a}, \mathtt{c}), (\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$;
- $\Lambda_2 = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{(\mathtt{a}, \mathtt{c}), (\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$; and
- $\Lambda_3 = \langle\{\mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{(\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$.

For $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$, Λ_1 (resp. Λ_2; and Λ_3) has only one extension $E_1 = \{\mathtt{a}, \mathtt{d}\}$ (resp. $E_2 = \{\mathtt{a}, \mathtt{b}, \mathtt{d}\}$, $E_3 = \{\mathtt{b}, \mathtt{d}\}$). Thus, $\mathtt{a}, \mathtt{b}, \mathtt{d}$ are possibly credulously/skeptically accepted, while only $\mathtt{d}$ is necessarily credulously/skeptically accepted. □

6.2 Relationship between EFAF and iAF

We start showing that every iAF can be rewritten into an extensions-equivalent EFAF, modulo meta-elements. The equivalence between an iAF $\Psi = \langle A, B, R, T\rangle$ and an EFAF $\Delta = \langle A', R', F, C\rangle$ derived from Ψ is in the sense that (i) $A \cup B \subseteq A'$ and $R \cup T \subseteq R'$ (i.e. arguments and attacks in Ψ also occur in Δ, but Δ may contain additional meta-arguments and meta-attacks), and (ii) for every completion $\Lambda \in comp(\Psi)$ there exists an s-subframework $\Lambda' \in \mathcal{S}(\Delta)$ (resp. c-subframework $\Lambda' \in \mathcal{C}(\Delta)$) such that $\sigma(\Lambda) = \sigma(\Lambda') \sqcap A \cup B$, and vice versa. Such equivalence under the set subframework (resp. cardinality subframework) semantics between an iAF Ψ and a (derived) EFAF Δ is denoted by $\Psi \approxeq_s \Delta$ (resp. $\Psi \approxeq_c \Delta$).

Theorem 4. *For any iAF $\Psi = \langle A, B, R, T\rangle$, the EFAF $\Delta_\Psi = \langle A_\Psi, R_\Psi, \emptyset, c_1 \wedge c_2 \wedge c_3\rangle$ is such that both $\Psi \approxeq_s \Delta_\Psi$ and $\Psi \approxeq_c \Delta_\Psi$ hold, with $A_\Psi = A \cup B \cup \{\overline{b} \,|\, b \in B\} \cup \{\overline{ab} \,|\, (a, b) \in R \wedge (\{a, b\} \cap B \neq \emptyset)\} \cup \{\overline{ab} \,|\, (a, b) \in T\}$, $R_\Psi = R \cup T$; and*

$$c_1: \bigwedge_{a \in A} arg(a) \wedge \bigwedge_{(a,b) \in R,\ \{a,b\} \subseteq A} att(a, b);$$

$$\begin{aligned} c_2: \bigwedge_{a \in B} (arg(a) \oplus arg(\overline{a})) \wedge \bigwedge_{(a,b) \in T} att(a, b) \oplus arg(\overline{ab}) \ \wedge \\ \bigwedge_{(a,b) \in R,\ \{a,b\} \cap B \neq \emptyset} att(a, b) \oplus arg(\overline{ab}); \end{aligned}$$

$$\begin{aligned} c_3: \bigwedge_{(a,b) \in R,\ a \in B,\ b \in A} &(arg(a) \equiv att(a, b)) &\wedge \\ \bigwedge_{(a,b) \in R,\ a \in A,\ b \in B} &(arg(b) \equiv att(a, b)) &\wedge \\ \bigwedge_{(a,b) \in R,\ \{a,b\} \subseteq B} &(arg(a) \wedge arg(b) \equiv att(a, b)) &\wedge \\ \bigwedge_{(a,b) \in T} &(att(a, b) \Rightarrow arg(a) \wedge arg(b)). \end{aligned}$$

Herein, c_1 states that certain arguments and certain attacks must belong to all subframeworks. Moreover, c_2 states that for any uncertain argument $a \in B$ (resp. attack $(a, b) \in T$), exactly one between a and $\overline{a}$ (resp. between $att(a, b)$ and $arg(\overline{ab})$) appears in any subframework (in the formulae above, $\oplus$ denotes the XOR operator); the same condition holds for the certain attacks (a, b) such that at least one of its arguments is uncertain in Ψ. Finally, c_3 states that certain attacks (a, b) of Ψ such that either a or b is uncertain must belong to a subframework iff both a and b belong to it; also, if an uncertain attack $(a, b) \in T$ belongs to a subframework then both arguments a and b must belong to it as well.

Example 8. Consider the iAF Ψ of Example 7 (see Figure 2). The EFAF derived from Ψ is $\Delta_\Psi = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \overline{\mathtt{a}}, \overline{\mathtt{ab}}, \overline{\mathtt{ac}}\}, R \cup T, \emptyset, c_1 \wedge c_2 \wedge c_3\rangle$ where:
$c_1 = arg(\mathtt{b}) \wedge arg(\mathtt{c}) \wedge arg(\mathtt{d}) \wedge att(\mathtt{b}, \mathtt{c}) \wedge att(\mathtt{c}, \mathtt{d})$;
$c_2 = (arg(\mathtt{a}) \oplus arg(\overline{\mathtt{a}})) \wedge (att(\mathtt{a}, \mathtt{b}) \oplus arg(\overline{\mathtt{ab}})) \wedge (att(\mathtt{a}, \mathtt{c}) \oplus arg(\overline{\mathtt{ac}}))$;
$c_3 = arg(\mathtt{a}) \equiv att(\mathtt{a}, \mathtt{c})$;

Δ_Ψ has 3 s-subframeworks (being also c-subframeworks):
$\Lambda'_1 = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}, \{(\mathtt{a}, \mathtt{b}), (\mathtt{b}, \mathtt{c}), (\mathtt{a}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$,
$\Lambda'_2 = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \overline{\mathtt{ab}}\}, \{(\mathtt{b}, \mathtt{c}), (\mathtt{a}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$,
$\Lambda'_3 = \langle\{\overline{\mathtt{a}}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \overline{\mathtt{ab}}, \overline{\mathtt{ac}}\}, \{(\mathtt{b}, \mathtt{c}), (\mathtt{c}, \mathtt{d})\}\rangle$,

It it easy to see that the set of σ-extensions (with $\sigma \in \{\mathtt{gr}, \mathtt{co}, \mathtt{st}, \mathtt{pr}\}$) of every Λ'_i is the same as that of Λ_i (of Example 7), modulo the meta-arguments $\overline{\mathtt{a}}$, $\overline{\mathtt{ab}}$, and $\overline{\mathtt{bc}}$, introduced in the rewriting, for $i \in [1, 3]$. □

6.3 Constrained and Correlated iAF

Two extensions of iAF related to our proposals are *correlated iAF* [Fazzinga *et al.*, 2021a; Fazzinga *et al.*, 2021b] and *constrained iAF* [Mailly, 2020]. Both frameworks extend iAF by adding a propositional formula which must be satisfied by completions, so that instead of considering the whole set of completions, only the ones satisfying the formula (called *valid completions*) are taken. The main difference between the two proposals is that while correlated iAF considers different fragments of propositional logic (PL), with the aim of studying complexity and expressivity of those fragments, constrained iAF considers general PL formulae. As propositional formulae can be rewritten using minimal sets of operators (e.g. $\wedge$ and $\neg$, or $\vee$ and $\neg$), the two proposals are equivalent in the sense that, although constrained iAF is more general, there are fragments of correlated iAF that have the same expressivity of general PL. Thus, hereafter we focus on constrained iAF.

Let γ_A denote the propositional language built over the propositional atoms in $\{arg(a) \mid a \in A\} \cup \{att(a, b) \mid a, b, \in A\}$, a constrained iAF can be defined as follows.

Definition 10. *A Constrained iAF (ciAF) is a tuple $\Psi = \langle A, B, R, T, C\rangle$, where $\langle A, B, R, T\rangle$ is an iAF, and $C \in \gamma_{A \cup B}$ is a propositional formula (called constraint).*

The constraint C is used to select a subset of the completions of iAF Ψ. The *valid completions* are those which satisfy C. The set of valid completions of Ψ is denoted by $valid\text{-}comp(\Psi)$. Extending the equivalence relations $\approxeq_s$ and $\approxeq_c$ to ciAF in the obvious way, we have the following.

Theorem 5. *For any ciAF $\Psi = \langle A, B, R, T, C\rangle$, the EFAF $\Delta_\Psi = \langle A_\Psi, R_\Psi, \emptyset, c_1 \wedge c_2 \wedge c_3 \wedge C\rangle$ is such that both $\Psi \approxeq_s \Delta_\Psi$ and $\Psi \approxeq_c \Delta_\Psi$ hold, where A_Ψ, R_Ψ, c_1, c_2, and c_3 are as given in Theorem 4.*

Example 9. Consider the ciAF $\Psi = \langle A, B, R, T, C\rangle$ where $\langle A, B, R, T\rangle$ is the iAF of Example 7, and $C = arg(\mathtt{a}) \Rightarrow att(\mathtt{a}, \mathtt{b})$. The EFAF derived from Ψ is $\Delta_\Psi = \langle\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \overline{\mathtt{a}}, \overline{\mathtt{ab}}, \overline{\mathtt{ac}}\}, R \cup T, \emptyset, c_1 \wedge c_2 \wedge c_3 \wedge C\rangle$ where c_is are given in Example 8. The subframeworks of Δ_Ψ are Λ'_1 and Λ'_3 given in Example 8, which correspond one-to-one to the valid completions of ciAF Ψ, modulo meta-elements. □

7 Related Work

We have already mentioned the relationships between EFAF and KBs repairs [Arenas *et al.*, 1999; Bertossi, 2011; Calautti *et al.*, 2022a; Calautti *et al.*, 2022b]. Computing repairs for AFs which are inconsistent, in the sense that they do not have any accepted argument, has been investigated in [Ulbricht and Baumann, 2019], where restoring consistency is achieved via dropping a minimal set of arguments or attacks. A similar approach has been defined for Assumption-Based Argumentation Frameworks in [Rapberger and Ulbricht, 2024]. In [Takahashi and Miwa, 2023] it is discussed how to make an AF with no stable extensions into one with a stable extension by adding a new argument (called 'repair').

The connection between Consistent Query Answering for DBs/KBs and argumentation has been investigated in [Croitoru and Vesic, 2013; Mahmood *et al.*, 2025; Bienvenu and Bourgaux, 2020; Mahmood *et al.*, 2024]. [Mahmood *et al.*, 2025] indicates that classical AF semantics can be reduced to DB repairs with functional and inclusion dependencies. It would be interesting to build on [Mahmood *et al.*, 2025] to map FAF to DB, thus storing additional argument features in such a DB and emulating subframeworks.

Further related approaches in AF concern those addressing the so-called *enforcing problem* and *minimal change problem* dealing with the question of whether it is possible to add (minimal) new information so that a desired set of arguments becomes an extension or at least a subset of one [Baumann, 2012; Kim *et al.*, 2013; Wallner *et al.*, 2017]. These works are all based on the addition of new information. We do not allow modifications that result in the addition of arguments and attacks, but we are interested in (minimal) removals ensuring satisfaction of user-defined constraints. Finally, the impact of adding or removing an argument on the set of extensions has been studied in [Cayrol *et al.*, 2010].

Extending AF by associating a claim to each argument representing its conclusion has been investigated in [Dvorák and Woltran, 2020; Dvorák *et al.*, 2023] where claim-augmented AFs are introduced. A claim-augmented AF can be modeled by a FAF whose set of facts contains assertions associating arguments to claims. Two approaches have been considered for defining claim-augmented AF semantics. The *inherited* semantics considers the extensions of the underlying AF before interpreting arguments by their claims, while the *claim-level* semantics redefines argumentation semantics in terms of the arguments' claims. We follow the inherited approach in defining FAF semantics, as often done in structured argumentation [Modgil and Prakken, 2014; Bondarenko *et al.*, 1993]. As a result, FAF generalizes claim-augmented AF under *inherited* semantics, as backed by our complexity analysis.

Interestingly, also Preference-based AF (PAF) under the preference reduction approach [Amgoud and Cayrol, 2002; Amgoud and Vesic, 2014; Kaci *et al.*, 2021; Bernreiter *et al.*, 2024a; Bernreiter *et al.*, 2024b] can be reduced to EFAF where PAF's *critical* attacks are tackled similarly to VAF's *unsuccessful* attacks [Dunne and Bench-Capon, 2004], as shown in what follows. Any VAF $\langle A, R, V, \text{VAL}, \text{VALPREF}\rangle$ can be seen as an EFAF $\langle A, R, F, C\rangle$ s.t. $F = \{value(a, \text{VAL}(a)) \mid a \in A\} \cup \{pref(u, v) \mid \{u, v\} \subseteq V \wedge \text{VALPREF}(u, v)\}$, and $C = \forall x, y, u, v.\, att(x, y) \wedge value(x, u) \wedge value(y, v) \Rightarrow \neg pref(v, u)$. This way the only subframework corresponds to the AF without unsuccessful attacks, thus emulating VAF semantics. Conversely, VAF cannot encode general FOL constraints of EFAF, as backed by the complexity analysis.

8 Conclusion

We have introduced the Featured AF, which extends AF by allowing arguments' features represented as a set of facts and queries allowing us to ask general questions concerning both arguments and features. We have further extended the framework by introducing EFAF that allows FOL constraints to be satisfied by subframeworks. For several argumentation semantics, EFAF turns out to be more expressive that iAF-based frameworks such as correlated and constrained AF. In a sense, EFAF can be seen as a general formalism that allows us to express features of well-known frameworks extending AF, such as iAF, correlated and constrained iAF, as well as features that cannot be expressed in AF or even in the extending frameworks, as backed by our complexity analysis. As future work we plan to extend our investigation to other semantics, such as semi-stable [Caminada, 2006] and to further extend the framework by also allowing the use of polynomial time functions in formulae (in both queries and constraints).

Acknowledgments

We acknowledge financial support from PNRR MUR projects FAIR (PE0000013) and SERICS (PE00000014), project Tech4You (ECS0000009), and MUR project PRIN 2022 EPICA (H53D23003660006).

References

[Alfano *et al.*, 2023a] G. Alfano, M. Calautti, S. Greco, F. Parisi, and I. Trubitsyna. Explainable acceptance in probabilistic and incomplete abstract argumentation frameworks. *Artif. Intell.*, 323:103967, 2023.

[Alfano *et al.*, 2023b] G. Alfano, S. Greco, F. Parisi, and I. Trubitsyna. Abstract argumentation framework with conditional preferences. In *Proc. of AAAI Conf.*, pages 6218–6227, 2023.

[Alfano *et al.*, 2023c] G. Alfano, S. Greco, F. Parisi, and I. Trubitsyna. Preferences and constraints in abstract argumentation. In *Proc. of International Joint Conference on Artificial Intelligence*, pages 3095–3103, 2023.

[Alfano *et al.*, 2024a] G. Alfano, A. Cohen, S. Gottifredi, S. Greco, F. Parisi, and G. R. Simari. Credulous acceptance in high-order argumentation frameworks with necessities: An incremental approach. *Artif. Intell.*, 333:104159, 2024.

[Alfano *et al.*, 2024b] G. Alfano, S. Greco, D. Mandaglio, F. Parisi, and I. Trubitsyna. Abstract argumentation frameworks with strong and weak constraints. *Artif. Intell.*, 336:104205, 2024.

[Amgoud and Cayrol, 1998] L. Amgoud and C. Cayrol. On the acceptability of arguments in preference-based argumentation. In *Proc. of UAI Conf.*, pages 1–7, 1998.

[Amgoud and Cayrol, 2002] L. Amgoud and C. Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *J. Autom. Reason.*, 29(2):125–169, 2002.

[Amgoud and Vesic, 2014] L. Amgoud and S. Vesic. Rich preference-based argumentation frameworks. *Int. J. Approx. Reason.*, 55(2):585–606, 2014.

[Arenas *et al.*, 1999] M. Arenas, L. E. Bertossi, and J. Chomicki. Consistent query answers in inconsistent databases. In *Proc. of PODS Conf.*, pages 68–79, 1999.

[Arieli, 2015] O. Arieli. Conflict-free and conflict-tolerant semantics for constrained argumentation frameworks. *J. Appl. Log.*, 13(4):582–604, 2015.

[Atkinson *et al.*, 2017] K. Atkinson, P. Baroni, M. Giacomin, A. Hunter, H. Prakken, C. Reed, G. R. Simari, M. Thimm, and S. Villata. Towards artificial argumentation. *Artificial Intelligence Magazine*, 38(3):25–36, 2017.

[Baumann, 2012] R. Baumann. What does it take to enforce an argument? minimal change in abstract argumentation. In *Proc. of ECAI Conf.*, pages 127–132, 2012.

[Baumeister *et al.*, 2018] D. Baumeister, D. Neugebauer, J. Rothe, and H. Schadrack. Verification in incomplete argumentation frameworks. *Artif. Intell.*, 264:1–26, 2018.

[Baumeister *et al.*, 2021] D. Baumeister, M. Järvisalo, D. Neugebauer, A. Niskanen, and J. Rothe. Acceptance in incomplete argumentation frameworks. *Artif. Intell.*, page 103470, 2021.

[Bench-Capon and Dunne, 2007] T. Bench-Capon and P. E. Dunne. Argumentation in artificial intelligence. *Artif. Intell.*, 171:619 – 641, 2007.

[Bernreiter *et al.*, 2024a] M. Bernreiter, W. Dvorák, A. Rapberger, and S. Woltran. The effect of preferences in abstract argumentation under a claim-centric view. *J. Artif. Intell. Res.*, 81:203–262, 2024.

[Bernreiter *et al.*, 2024b] M. Bernreiter, W. Dvorák, and S. Woltran. Abstract argumentation with conditional preferences. *Argument Comput.*, 15(2):161–189, 2024.

[Bertossi, 2011] L. E. Bertossi. *Database Repairing and Consistent Query Answering*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.

[Bienvenu and Bourgaux, 2020] M. Bienvenu and C. Bourgaux. Querying and repairing inconsistent prioritized knowledge bases: Complexity analysis and links with abstract argumentation. In *Proc. of KR Conf.*, pages 141–151, 2020.

[Bondarenko *et al.*, 1993] A. Bondarenko, F. Toni, and R. A. Kowalski. An assumption-based framework for non-monotonic reasoning. In *Proc. of LPNMR Workshop*, pages 171–189, 1993.

[Brewka *et al.*, 2013] G. Brewka, H. Strass, S. Ellmauthaler, J. P. Wallner, and S. Woltran. Abstract dialectical frameworks revisited. In *Proc. of IJCAI Conf.*, pages 803–809, 2013.

[Calautti *et al.*, 2022a] M. Calautti, S. Greco, C. Molinaro, and I. Trubitsyna. Preference-based inconsistency-tolerant query answering under existential rules. *Artif. Intell.*, 312:103772, 2022.

[Calautti *et al.*, 2022b] M. Calautti, S. Greco, C. Molinaro, and I. Trubitsyna. Query answering over inconsistent knowledge bases: A probabilistic approach. *Theor. Comp. Sci.*, 935:144–173, 2022.

[Caminada, 2006] M. Caminada. Semi-stable semantics. In *Proc. of COMMA Conf.*, pages 121–130, 2006.

[Cayrol *et al.*, 2010] C. Cayrol, F. D. de Saint-Cyr, and M. Lagasquie-Schiex. Change in abstract argumentation frameworks: Adding an argument. *J. Artif. Intell. Res.*, 38:49–84, 2010.

[Cayrol *et al.*, 2018] C. Cayrol, J. Fandinno, L. F. del Cerro, and M. Lagasquie-Schiex. Structure-based semantics of argumentation frameworks with higher-order attacks and supports. In *Proc. of COMMA Conf.*, pages 29–36, 2018.

[Cohen *et al.*, 2015] A. Cohen, S. Gottifredi, A. J. Garcia, and G. R. Simari. An approach to abstract argumentation with recursive attack and support. *J. Appl. Log.*, 13(4):509–533, 2015.

[Coste-Marquis *et al.*, 2006] S. Coste-Marquis, C. Devred, and P. Marquis. Constrained argumentation frameworks. In *Proc. of KR Conf.*, pages 112–122, 2006.

[Croitoru and Vesic, 2013] M. Croitoru and S. Vesic. What can argumentation do for inconsistent ontology query answering? In *Proc. of SUM Conf.*, volume 8078, pages 15–29, 2013.

[Dimopoulos and Torres, 1996] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comp. Sci.*, 170(1-2):209–244, 1996.

[Dimopoulos *et al.*, 2019] Y. Dimopoulos, J. Mailly, and P. Moraitis. Argumentation-based negotiation with incomplete opponent profiles. In *Proc. of the AAMAS Conf.*, pages 1252–1260, 2019.

[Dung and Thang, 2010] P. M. Dung and P. M. Thang. Towards (probabilistic) argumentation for jury-based dispute resolution. In *Proc. of COMMA Conf.*, pages 171–182, 2010.

[Dung, 1995] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77:321–358, 1995.

[Dunne and Bench-Capon, 2002] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artif. Intell.*, 141(1/2):187–203, 2002.

[Dunne and Bench-Capon, 2004] P. E. Dunne and T. J. M. Bench-Capon. Complexity in value-based argument systems. In *Proc. of JELIA Conf.*, pages 360–371, 2004.

[Dvorak and Dunne, 2017] W. Dvorak and P. E. Dunne. Computational problems in formal argumentation and their complexity. *FLAP*, 4(8), 2017.

[Dvorák and Woltran, 2020] W. Dvorák and S. Woltran. Complexity of abstract argumentation under a claim-centric view. *Artif. Intell.*, 285:103290, 2020.

[Dvorák *et al.*, 2023] W. Dvorák, A. Greßler, A. Rapberger, and S. Woltran. The complexity landscape of claim-augmented argumentation frameworks. *Artif. Intell.*, 317:103873, 2023.

[Fazzinga *et al.*, 2021a] B. Fazzinga, S. Flesca, and F. Furfaro. Reasoning over argument-incomplete aafs in the presence of correlations. In *Proc. of IJCAI Conf.*, pages 189–195, 2021.

[Fazzinga *et al.*, 2021b] B. Fazzinga, S. Flesca, and F. Furfaro. Reasoning over attack-incomplete aafs in the presence of correlations. In *Proc. of KR Conf.*, pages 301–311, 2021.

[Fazzinga *et al.*, 2022a] B. Fazzinga, S. Flesca, and F. Furfaro. Abstract argumentation frameworks with marginal probabilities. In *Proc. of IJCAI*, pages 2613–2619, 2022.

[Fazzinga *et al.*, 2022b] B. Fazzinga, S. Flesca, F. Furfaro, and L. Pontieri. Process mining meets argumentation: Explainable interpretations of low-level event logs via abstract argumentation. *Inf. Syst.*, 107:101987, 2022.

[Gottifredi *et al.*, 2018] S. Gottifredi, A. Cohen, A. J. Garcia, and G. R. Simari. Characterizing acceptability semantics of argumentation frameworks with recursive attack and support relations. *Artif. Intell.*, 262:336–368, 2018.

[Greco *et al.*, 2001] G. Greco, S. Greco, and E. Zumpano. A logic programming approach to the integration, repairing and querying of inconsistent databases. In *Proc. of ICLP Conf.*, pages 348–364, 2001.

[Hunter, 2012] A. Hunter. Some foundations for probabilistic abstract argumentation. In *Proc. of COMMA Conf.*, pages 117–128, 2012.

[Kaci *et al.*, 2021] S. Kaci, L. W. N. van der Torre, S. Vesic, and S. Villata. Preference in abstract argumentation. In *Handbook of Formal Argumentation*, volume 2, chapter 3. College Publications, 2021.

[Kim *et al.*, 2013] E. J. Kim, S. Ordyniak, and S. Szeider. The complexity of repairing, adjusting, and aggregating of extensions in abstract argumentation. In *Proc. of TAFA Workshop*, pages 158–175, 2013.

[Li *et al.*, 2011] H. Li, N. Oren, and T. J. Norman. Probabilistic argumentation frameworks. In *Proc. of TAFA Workshop*, pages 1–16, 2011.

[Mahmood *et al.*, 2024] Y. Mahmood, J. Virtema, T. Barlag, and A. N. Ngomo. Computing repairs under functional and inclusion dependencies via argumentation. In *Proc. of FoIKS Symp.*, volume 14589, pages 23–42, 2024.

[Mahmood *et al.*, 2025] Y. Mahmood, M. Hecher, and A. N. Ngomo. Dung's argumentation framework: Unveiling the expressive power with inconsistent databases. In *Proc. of AAAI Conf.*, pages 15058–15066, 2025.

[Mailly, 2020] J. Mailly. Possible controllability of control argumentation frameworks. In *Proc. of COMMA Conf.*, pages 283–294, 2020.

[Mailly, 2024] J. Mailly. Constrained incomplete argumentation frameworks: Expressiveness, complexity and enforcement. *AI Commun.*, 37(3):299–322, 2024.

[Modgil and Prakken, 2013] S. Modgil and H. Prakken. A general account of argumentation with preferences. *Artif. Intell.*, 195:361–397, 2013.

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

[Nouioua and Risch, 2011] F. Nouioua and V. Risch. Argumentation frameworks with necessities. In *Proc. of SUM Conf.*, 2011.

[Nouioua, 2013] F. Nouioua. Afs with necessities: Further semantics and labelling characterization. In *Proc. of SUM Conf.*, pages 120–133, 2013.

[Prakken, 2009] H. Prakken. Models of persuasion dialogue. In *Argumentation in AI*, pages 281–300. 2009.

[Rapberger and Ulbricht, 2024] A. Rapberger and M. Ulbricht. Repairing assumption-based argumentation frameworks. In *Proc. of KR Conf.*, 2024.

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

[Takahashi and Miwa, 2023] K. Takahashi and H. Miwa. Topological conditions and solutions for repairing argumentation frameworks. In *Proc. of CLAR Conf.*, pages 101–118, 2023.

[Ulbricht and Baumann, 2019] M. Ulbricht and R. Baumann. If nothing is accepted - repairing argumentation frameworks. *J. Artif. Intell. Res.*, 66:1099–1145, 2019.

[Villata *et al.*, 2012] S. Villata, G. Boella, D. M. Gabbay, and L. W. N. van der Torre. Modelling defeasible and prioritized support in bipolar argumentation. *Ann. Math. Artif. Intell.*, 66(1-4), 2012.

[Wallner *et al.*, 2017] J. P. Wallner, A. Niskanen, and M. Järvisalo. Complexity results and algorithms for extension enforcement in abstract argumentation. *J. Artif. Intell. Res.*, 60:1–40, 2017.