ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-551

Synthesizing Argumentation Frameworks from Examples

Andreas Niskanen and **Johannes P. Wallner** and **Matti Järvisalo**[1]

Abstract. Argumentation is nowadays a core topic in AI research. Understanding computational and representational aspects of abstract argumentation frameworks (AFs) is a central topic in the study of argumentation. The study of realizability of AFs aims at understanding the expressive power of AFs under different semantics. We propose and study the *AF synthesis problem* as a natural extension of realizability, addressing some of the shortcomings arising from the relatively stringent definition of realizability. Specifically, AF synthesis seeks to construct, or synthesize, AFs that are semantically closest to the knowledge at hand even when no AFs exactly representing the knowledge exist. Going beyond defining the AF synthesis problem, we (i) prove NP-completeness of AF synthesis under several semantics, (ii) study basic properties of the problem in relation to realizability, (iii) develop algorithmic solutions to AF synthesis using constrained optimization, (iv) empirically evaluate our algorithms on different forms of AF synthesis instances, as well as (v) discuss variants and generalization of AF synthesis.

1 INTRODUCTION

The study of representational and computational aspects of argumentation is a core topic in modern artificial intelligence (AI) research [5]. A current strong focus of argumentation research is the extension-based setting of abstract argumentation frameworks (AFs) [14] and its generalizations. A fundamental knowledge representational aspect related to AFs is *realizability* [15], i.e., the question of whether a specific AF semantics allows for exactly representing a given set of extensions as an AF. With important motivations from various perspectives—including the analysis of the relationships of central AF semantics [15] (in terms of the range of sets of extensions different semantics allow for representing as AFs) and connections to the study of argumentation dynamics [13, 11] (in terms of the ability to construct an AF for revised extensions)—realizability has recently been studied by several authors [15, 3, 16, 23, 19, 20].

While the study of realizability has provided various fundamental insights into AFs, the concept of realizability is quite strict in that a set E of extensions is considered realizable (under a specific AF semantics σ) if and only if there is an AF the σ-extensions of which are *exactly* those in E. Implicitly, this definition hence requires that *all* other sets of arguments *must not* be extensions of the AF of interest. This strictness requires that we have *complete* knowledge of the extensions of interest, and further, in order to actually construct a corresponding AF of interest, relies on the assumption that the set of extensions are *not conflicting* in terms of allowing them to be exactly represented by an AF. However, from more practical perspectives, we foresee these requirements to be somewhat cumbersome. Firstly, the requirement of complete knowledge implies in the worst case taking into account an exponential number of extensions. Secondly, the definition does not allow for "mistakes" or noise in the process of obtaining the extensions, and also rules out the possibility of dealing with multiple sources of potentially conflicting sets of extensions.

In this work, with a central goal of generalizing the concept of realizability to accommodate incomplete and noise information on extensions, we propose and study what we call the *AF synthesis problem*[2]. Specifically, AF synthesis relaxes the notion of realizability to incomplete information—assuming only partial knowledge of extensions and non-extensions as *positive* and *negative examples*—and noisy settings, by allowing for expressing relative trust in the examples via weights. In this generalized setting, we define AF synthesis as the constrained optimization task of finding an AF that optimally represents the given examples in terms of minimizing the costs (defined via the weights of the given examples) incurred from the AF by including a negative example or not including a positive example. Beyond precisely defining the AF synthesis problem, our main contributions include the following.

- We formally analyze the relationship of AF synthesis and realizability in terms of necessary and sufficient conditions for an AF synthesis instance to be realizable under different AF semantics (Section 3).
- We provide complexity results for AF synthesis under three central AF semantics, namely, the conflict-free, admissible, and stable semantics, with the main result that AF synthesis is in the general case NP-complete under each of these semantics (Section 4).
- We develop a first constraint-based approach to optimal AF synthesis, by providing declarative encodings for AF synthesis in the Boolean optimization paradigm of maximum satisfiability (MaxSAT), and furthermore, discuss how by simple modifications to the encoding one can account for structural constraints over the AFs to be synthesized (Section 5).
- We present results from an empirical evaluation of the approach based on benchmarks from the recent ICCMA'15 argumentation solver competition [26] as well as additional randomly generated AF synthesis instances (Section 6).
- We discuss further variants and generalizations of AF synthesis from representational and computational complexity perspectives, including how to adapt the problem to multiple sources of extensions and allowing for mixtures of different AF semantics, as well as symbolic representations of examples via Boolean formulas (Section 7).

A more detailed overview of connections to related work is provided after the main contributions (Section 8). For readability, more complicated formal proofs are detailed in Appendix A.

[1] Helsinki Institute for Information Technology HIIT, Department of Computer Science, University of Helsinki, Finland

[2] Alternatively, one could refer to the problem focused on in this paper as an AF *learning* problem.

2 ARGUMENTATION FRAMEWORKS

We start by briefly recalling argumentation frameworks [14] (see also [2]) as the central formalism in abstract argumentation.

Definition 1. *An* argumentation framework (AF) *is a pair $F = (A, R)$, where A is a finite non-empty set of arguments and $R \subseteq A \times A$ is the attack relation. The pair $(a, b) \in R$ indicates that a attacks b. An argument $a \in A$ is* defended *(in F) by a set $S \subseteq A$ if, for each $b \in A$ such that $(b, a) \in R$, there is a $c \in S$ such that $(c, b) \in R$.*

Semantics for AFs are defined through functions σ which assign to each AF $F = (A, R)$ a set $\sigma(F) \subseteq 2^A$ of extensions. We consider for σ the functions stb, adm, com, and grd, which stand for stable, admissible, complete, and grounded, respectively.

Definition 2. *Given an AF $F = (A, R)$, the* characteristic function *$\mathcal{F}_F : 2^A \to 2^A$ of F is $\mathcal{F}_F(S) = \{x \in A \mid x \text{ is defended by } S\}$. Moreover, for a set $S \subseteq A$, the* range of S *is $S_R^+ = S \cup \{x \mid (y, x) \in R, y \in S\}$.*

Definition 3. *Let $F = (A, R)$ be an AF. A set $S \subseteq A$ is* conflict-free *(in F) if there are no $a, b \in S$ such that $(a, b) \in R$. We denote the collection of conflict-free sets of F by $cf(F)$. For a conflict-free set $S \in cf(F)$ it holds that*

- *$S \in stb(F)$ iff $S_R^+ = A$;*
- *$S \in adm(F)$ iff $S \subseteq \mathcal{F}_F(S)$;*
- *$S \in com(F)$ iff $S = \mathcal{F}_F(S)$;*
- *$S \in grd(F)$ iff S is the least fixed-point of $\mathcal{F}_F$.*

For any AF F, we have $cf(F) \supseteq adm(F) \supseteq com(F) \supseteq stb(F)$. We use "$\sigma$-extension" to denote an extension under a semantics σ.

3 THE AF SYNTHESIS PROBLEM

In this section we introduce the AF synthesis problem. For a given set of weighted examples that represent semantical information, with weights intuitively representing relative trust in the examples, the task is to synthesize an AF that has minimum cost over the examples not satisfied. We assume a given non-empty set of arguments A from which we are to construct an AF. Formally, an example $e = (S, w)$ is a pair with S a subset of the set of arguments, i.e., $S \subseteq A$, and a positive integer $w > 0$ representing the example's weight. We denote the set of arguments of an example $e = (S, w)$ by $S_e = S$ and weight by $w_e = w$. For a set E of examples, we define $\mathbb{S}_E = \{S_e \mid e \in E\}$ as a shorthand for the set of all sets of arguments occurring in E.

An instance of the AF synthesis problem is a quadruple $P = (A, E^+, E^-, \sigma)$, with a non-empty set A of arguments, two sets of examples, E^+ and E^-, that we call positive and negative examples, respectively, and semantics σ. An AF F satisfies a positive example e if $S_e \in \sigma(F)$; similarly, F satisfies a negative example if $S_e \notin \sigma(F)$. For a given AF F, the associated cost w.r.t. P, denoted by $cost(P, F)$, is the sum of weights of examples not satisfied by F. Formally, $cost(P, F)$ is

$$\sum_{e \in E^+} w_e \cdot I(S_e \notin \sigma(F)) + \sum_{e' \in E^-} w_{e'} \cdot I(S_{e'} \in \sigma(F)),$$

where $I(\cdot)$ is the indicator function that returns 1 if the property (membership in a set) is satisfied, and otherwise 0. The task in AF synthesis is to find an AF of minimum cost over all AFs.

AF Synthesis
INPUT: $P = (A, E^+, E^-, \sigma)$
TASK: Find an AF F^* with

$$F^* \in \underset{F=(A,R)}{\arg\min}(cost(P, F)).$$

Example 1. *Consider the set of positive examples $E^+ = \{(\{a, b\}, 1), (\{a, c\}, 1), (\{b, c\}, 5)\}$ and the set of negative examples $E^- = \{(\{a\}, 1), (\{a, b, c\}, 5)\}$. We illustrate these examples in Figure 1. Here we see that the positive examples claim together that each pair of arguments of A is a σ-extension, and the negative examples claim that the whole set A is not a σ-extension and that the singleton set $\{a\}$ is likewise not a σ-extension. Let $P_{cf} = (A, E^+, E^-, cf)$ with $A = \{a, b, c\}$ be an AF synthesis instance under conflict-free semantics. An optimal solution AF $F_{cf} = (A, R_{cf})$ with $cost(P_{cf}, F_{cf}) = 2$ is given by $R_{cf} = \{(a, b)\}$. This AF F_{cf} does not satisfy the positive example $(\{a, b\}, 1)$ and the negative example $(\{a\}, 1)$.*

Regarding admissible semantics, let $P_{adm} = (A, E^+, E^-, adm)$. In this case AF $F_{adm} = (A, R_{adm})$ is an optimal solution with $R_{adm} = \{(b, a)\}$. Except for positive examples $(\{a, b\}, 1)$ and $(\{a, c\}, 1)$, all other examples are satisfied by F_{adm} for P_{adm}. Thus $cost(P_{adm}, F_{adm}) = 2$.

For stable semantics, let $P_{stb} = (A, E^+, E^-, stb)$. An optimal solution AF to P_{stb} is given by $F_{stb} = (A, R_{stb})$ with $R_{stb} = \{(a, b), (b, a)\}$. Here $stb(F_{stb}) = \{\{a, c\}, \{b, c\}\}$ and $cost(P_{stb}, F_{stb}) = 1$.

We now investigate the existence of 0-cost solutions for the AF synthesis problem by relating the problem with realizability results from [15]. In contrast to the AF synthesis problem, [15] consider the problem of a given unweighted set $\mathbb{S}$ of sets of arguments, and ask whether there is an AF F s.t. $\mathbb{S} = \sigma(F)$. In words, in the setting of realizability, the given set exactly specifies which sets have to be σ-extensions and which must not be σ-extensions. Further, [15] do not consider weights attached to examples, and the set of arguments A is not specified and may contain more arguments than occurring in $\mathbb{S}$. Restricting the set of arguments to only arguments occurring in $\mathbb{S}$ is studied in [3, 20], although not directly applicable to our problem.

We make use of and generalize the notions proposed in [15] by specifying conditions under which 0-cost solutions exist and as well as properties 0-cost solutions satisfy. We first focus on the conflict-free semantics. We utilize the following concept adapted from [15, Definitions 6 and 7], defining a consequence operator that states which sets must be conflict-free if we assume a given set of sets $\mathbb{S}$ to be conflict-free in an AF. Let $\mathit{ImpliedCF}(\mathbb{S}) = \{X \mid a, b \in X \text{ implies } \exists S \in \mathbb{S} \text{ with } \{a, b\} \subseteq S\}$. Intuitively, if each set in $\mathbb{S}$ is conflict-free, and each pair of arguments in a set X is contained in one set of $\mathbb{S}$, then X is conflict-free as well. Note that a and b in this definition need not be distinct ($\{a, b\}$ is equal to $\{a\}$ if $a = b$). Further, $\emptyset$ is in $\mathit{ImpliedCF}(\mathbb{S})$ for any $\mathbb{S}$.

Lemma 1. *Let $F = (A, R)$ be an AF and $\mathbb{S} \subseteq 2^A$. If $\mathbb{S} \subseteq cf(F)$, then $\mathit{ImpliedCF}(\mathbb{S}) \subseteq cf(F)$.*

Example 2. *Continuing from Example 1, consider $\mathbb{S}_{E^+} = \{\{a, b\}, \{a, c\}, \{b, c\}\}$. If each element of $\mathbb{S}_{E^+}$ is conflict-free in an AF F ($\mathbb{S}_{E^+} \subseteq cf(F)$), then, e.g., $\{a, b, c\} \in cf(F)$, since there cannot be an attack between any of these three arguments. In particular, we have $\mathit{ImpliedCF}(\mathbb{S}_{E^+}) = \mathbb{S}_{E^+} \cup \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b, c\}\}$. This directly shows for P_{cf} from Example 1 that there is no solution*

positive E^+	negative E^-
$e_1 = (\{a, b\}, 1)$	$e_4 = (\{a\}, 1)$
$e_2 = (\{a, c\}, 1)$	$e_5 = (\{a, b, c\}, 5)$
$e_3 = (\{b, c\}, 5)$	

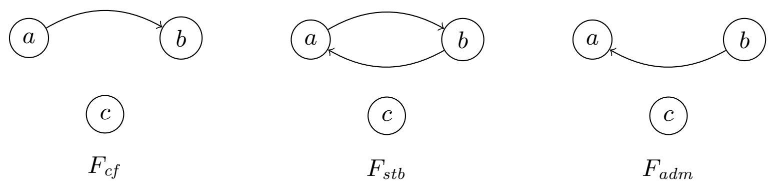

Figure 1. AF synthesis example with optimal solution AFs

AF to P_{cf} of cost 0. In fact, there is no AF satisfying both the positive example $(\{a, b\}, 1)$ and the negative example $(\{a\}, 1)$ under the conflict-free semantics. Also, there is no AF satisfying all three positive examples and negative example $(\{a, b, c\}, 5)$ under the conflict-free semantics.

Equipped with the preceding lemma, we give a necessary and sufficient condition for 0-cost solutions for AF synthesis under the conflict-free semantics.

Proposition 2. *Let $P = (A, E^+, E^-, cf)$ be an instance of AF synthesis. There is a solution AF F to P with $cost(P, F) = 0$ iff $ImpliedCF(\mathbb{S}_{E^+}) \cap \mathbb{S}_{E^-} = \emptyset$.*

Intuitively, to synthesize an AF F that has $\mathbb{S}_{E^+}$ as its conflict-free sets, $ImpliedCF(\mathbb{S}_{E^+})$ need to be conflict-free, too. Moreover, using results from [15], one can show that there is an AF F with $cf(F) = ImpliedCF(\mathbb{S}_{E^+})$. Furthermore, if no negative example e claims that a set of $ImpliedCF(\mathbb{S}_{E^+})$ should not be conflict-free, i.e., $S_e \notin ImpliedCF(\mathbb{S}_{E^+})$, then this implies that F has cost 0.

Now consider admissible sets. Similarly as for conflict-free sets, we define the following consequence operator. For a set of sets $\mathbb{S}$, let $ImpliedADM(\mathbb{S}) = \{X \mid X = \bigcup_{S \in \mathbb{S}'} S,\ \mathbb{S}' \subseteq \mathbb{S},\ X \in ImpliedCF(\mathbb{S})\}$. Briefly put, if we assume $\mathbb{S}$ to be a collection of admissible sets, then each union of sets in $\mathbb{S}$ that is conflict-free, i.e., in $ImpliedCF(\mathbb{S})$, is also an admissible set. By this definition, $\emptyset \in ImpliedADM(\mathbb{S})$ for any $\mathbb{S}$.

Lemma 3. *Let $F = (A, R)$ be an AF and $\mathbb{S} \subseteq 2^A$. If $\mathbb{S} \subseteq adm(F)$, then $ImpliedADM(\mathbb{S}) \subseteq adm(F)$.*

Example 3. *Consider $\mathbb{S}_{E^+} = \{\{a, b\}, \{a, c\}, \{b, c\}\}$ from Example 1. Then $ImpliedADM(\mathbb{S}_{E^+}) = \mathbb{S}_{E^+} \cup \{\{a, b, c\}, \emptyset\}$. Regarding the set $\mathbb{S}' = \{\{b, c\}\}$ which is the set of positive examples satisfied by F_{adm}, we have $ImpliedADM(\mathbb{S}') = \mathbb{S}' \cup \{\emptyset\}$.*

Consider $P'_{adm} = (\{a, b, c\}, E_1^+, E_1^-, adm)$ with $E_1^+ = \{(\{a, c\}, 1), (\{b, c\}, 1)\}$ and $E_1^- = \{(\{a\}, 1)\}$. We have $ImpliedADM(\mathbb{S}_{E_1^+}) = \mathbb{S}_{E_1^+} \cup \{\emptyset\}$ and $ImpliedADM(\mathbb{S}_{E_1^+}) \cap \mathbb{S}_{E_1^-} = \emptyset$. Unlike for the conflict-free semantics, this condition for the admissible semantics does not imply existence of a 0-cost solution AF for P'_{adm}. In fact, a 0-cost solution AF does not exist for P'_{adm}. A 0-cost solution is possible, however, if the set of arguments A includes more arguments. For instance, take $F' = (\{a, b, c, d\}, \{(b, a), (a, b), (c, d), (d, a), (d, d)\})$. We have $adm(F') = \{\emptyset, \{b\}, \{c\}, \{b, c\}, \{a, c\}\}$ and, for $P''_{adm} = (\{a, b, c, d\}, E_1^+, E_1^-, adm)$, we have $cost(P''_{adm}, F') = 0$. Such "auxiliary" arguments, i.e., arguments not present in the examples, are not always required for 0-cost solutions under the admissible semantics. For instance, given only two positive examples $(\{a, c\}, 1)$ and $(\{b, c\}, 1)$, and negative example $(\{a, b, c\}, 1)$, one can synthesize a 0-cost AF with a mutual attack between a and b.

Similarly as for conflict-free semantics, each 0-cost solution under the admissible semantics implies that for no negative examples e we have $S_e \in ImpliedADM(\mathbb{S}_{E^+})$. For existence of an AF F with $ImpliedADM(\mathbb{S}_{E^+}) = adm(F)$, we make use of results from [15] which requires auxiliary arguments, i.e., arguments not present in $\mathbb{S}_{E^+}$. We use the abstract function $AuxArgs(adm, \mathbb{S}_{E^+})$ that returns the number of auxiliary arguments needed to construct F as specified in [15, Definitions 13 and 14].

Proposition 4. *Let $P = (A, E^+, E^-, adm)$ be an instance of the AF synthesis problem. Consider the following conditions.*

1. *$ImpliedADM(\mathbb{S}_{E^+}) \cap \mathbb{S}_{E^-} = \emptyset$.*
2. *$|A \setminus (\bigcup_{S \in \mathbb{S}_{E^+}} S)| > AuxArgs(adm, \mathbb{S}_{E^+})$.*

If there is a solution AF F to P with $cost(P, F) = 0$, then condition 1 holds. If both conditions 1 and 2 hold, then there is a solution AF F to P with $cost(P, F) = 0$.

We move on to the stable semantics, under which the picture is more complex. Existence of a 0-cost solution for an AF synthesis instance implies that the set of positive examples $\mathbb{S}_{E^+}$ is $\subseteq$-incomparable, does not include $\emptyset$, is disjoint from the negative sets $\mathbb{S}_{E^-}$, and no positive set $S \in \mathbb{S}_{E^+}$ is a proper subset of an implied conflict-free set in $ImpliedCF(\mathbb{S}_{E^+})$. These conditions are quite intuitive, since, e.g., a violation of the last condition violates the fact that stable extensions attack all arguments outside the set.

These conditions imply the existence of 0-cost solutions if a certain number of auxiliary arguments is available in A, i.e., arguments not present in $\mathbb{S}_{E^+}$. For achieving this result, we use again results from [15], providing a construction in this case that utilizes such auxiliary arguments to synthesize the AF. We provide here a rough bound for auxiliary arguments from [15, Definition 12]. More concretely, we use the function $AuxArgs(stb, \mathbb{S}_{E^+})$ that is equal to the maximum number of stable extensions for any AF with $|\mathbb{S}_{E^+}|$ many arguments (for more details, see [4, Theorem 1]).

Proposition 5. *Let $P = (A, E^+, E^-, stb)$ be an instance of the AF synthesis problem. Consider the following conditions.*

1. *Any two distinct $S, S' \in \mathbb{S}_{E^+}$ are incomparable w.r.t. $\subseteq$.*
2. *$\forall S \in \mathbb{S}_{E^+}$ we have $S \not\subset S'$ for all $S' \in ImpliedCF(\mathbb{S}_{E^+})$.*
3. *$\emptyset \notin \mathbb{S}_{E^+}$.*
4. *$\mathbb{S}_{E^+} \cap \mathbb{S}_{E^-} = \emptyset$.*
5. *$|A \setminus (\bigcup_{S \in \mathbb{S}_{E^+}} S)| > AuxArgs(stb, \mathbb{S}_{E^+})$.*

If there is a solution AF F to P with $cost(P, F) = 0$, then conditions 1-4 hold. If conditions 1-5 hold, then there is a solution AF F to P with $cost(P, F) = 0$.

Interestingly, the negative examples play a relatively minor role in 0-cost solutions under the stable semantics (see condition 4 of Proposition 5). In contrast to conflict-free or admissible sets, existence of stable extensions does not directly imply existence of further stable extensions for an unrestricted set of arguments A (this observation is also implicitly stated in [15, Lemma 2 and Proposition 1]).

Example 4. *The AF F_{stb} from Example 1 has $stb(F_{stb}) = \{\{a,c\},\{b,c\}\} = \mathbb{S}'$. Conditions 1-4 from Proposition 5 hold for $\mathbb{S}'$. One can synthesize an AF, e.g., F_{stb}, as a 0-cost solution to $P'_{stb} = (\{a,b,c\}, \{(\{a,c\},1),(\{b,c\},1)\}, \emptyset, stb)$ without auxiliary arguments. An example where auxiliary arguments are required to synthesize an AF with 0-cost can be found in [3].*

4 COMPLEXITY OF AF SYNTHESIS

We continue by analyzing the computational complexity of the AF synthesis problem. As the main results of this section, we show that AF synthesis is NP-complete in the unrestricted case under the conflict-free, admissible, and stable semantics. Furthermore, we show that while restricting either E^+ or E^- to be empty yields fragments of the problem where a trivial AF solves the problem optimally, NP-completeness persists even for $E^- = \emptyset$ under the stable semantics. The results are summarized in Table 1.

Table 1. Complexity of AF synthesis

	no restrictions	$E^+ = \emptyset$	$E^- = \emptyset$
Conflict-free	NP-c	trivial	trivial
Admissible	NP-c	trivial	trivial
Stable	NP-c	trivial	NP-c

We first outline special cases of AF synthesis in which a trivial solution AF is guaranteed to be optimal. In particular, if no positive examples are present, then the complete digraph $F = (A, 2^A \times 2^A)$ satisfies all negative examples (F has no stable extensions, and the only conflict-free and admissible set is $\emptyset$). If the set of negative examples E^- is empty, then AF synthesis under the conflict-free and admissible semantics is trivial by constructing the AF $F = (A, \emptyset)$ (every subset of A is conflict-free and admissible).

Proposition 6 (Trivial solutions)**.** *An optimal solution AF F^* can be computed in polynomial time for an AF synthesis instance $P = (A, E^+, E^-, \sigma)$ if one of the following conditions holds.*

1. *$\sigma \in \{cf, adm, stb\}$ and $E^+ = \emptyset$.*
2. *$\sigma \in \{cf, adm\}$ and $E^- = \emptyset$.*

We now turn our attention to the NP-hard cases of the AF synthesis problem under the conflict-free, admissible, and stable semantics. Formally, the decision problem corresponding to AF synthesis consists of an AF synthesis instance $P = (A, E^+, E^-, \sigma)$ and an integer $k \geq 0$, and asks whether there is an AF $F = (A, R)$ with $cost(P, F) \leq k$.

Intuitively, the main source of NP-hardness for the AF synthesis problem for the considered semantics lies in finding an optimal subset of examples from which to synthesize an AF. We start with the conflict-free semantics and prove NP-hardness by a reduction from the Boolean satisfiability problem; recall that all formal proofs are detailed in Appendix A. For intuition on the reduction, "choosing" a truth assignment can be simulated by a set of examples similarly as in Example 1 (shown in Figure 1). In other words, for positive examples containing the sets $\{a,b\}$, $\{a,c\}$, and $\{b,c\}$, and negative example $\{a,b,c\}$, one can attach high weights (beyond the bound k) to the last two examples and unit weights to the first two. In this way any solution AF of cost at most k does not satisfy both of the unit-weighted examples, thus mimicking a truth assignment, i.e., one has to choose which of these two examples to satisfy. The reduction is completed by additional examples that simulate satisfaction of clauses of a Boolean formula, by ensuring that attacks have to be present via negative examples.

Proposition 7. *AF synthesis is* NP*-complete under the conflict-free semantics.*

The reduction we use for establishing NP-hardness under the admissible semantics follows essentially the same idea.

Proposition 8. *AF synthesis is* NP*-complete under the admissible semantics.*

For the stable semantics, we establish NP-completeness as well; however, surprisingly, in contrast to the conflict-free and admissible semantics, AF synthesis under the stable semantics is NP-complete even when E^- is empty. The reduction is technically more involved. Intuitively, presence of attacks can be simulated via arguments outside the set of a positive example, since if the set is stable, it has to attack all arguments outside the set. This is also a reason why hardness holds even if E^- is empty.

Proposition 9. *AF synthesis is* NP*-complete for stable semantics, even if the set of negative examples is empty.*

Finally we observe that AF synthesis under the grounded semantics is trivial, since exactly one grounded extension is present in an AF.

Proposition 10. *Let $P = (A, E^+, E^-, grd)$ be an instance of the AF synthesis problem. An optimal solution AF F^* to P can be constructed in polynomial time.*

Proof. For each $e \in E^+$ an AF F with $grd(F) = \{S_e\}$ and with $cost(P, F)$ equal to the sum of all weights of $E^+ \setminus \{e\}$ plus $w_{e'}$ if $e' \in E^-$ and $S_e = S_{e'}$, can be constructed in polynomial time by adding a self-attack to all $A \setminus S_e$. Further, if $2^A \setminus \mathbb{S}_{E^+}$ is non-empty, pick $S \in 2^A \setminus \mathbb{S}_{E^+}$ with minimum-weighted $e'' \in E^- \cup \{e \mid S_e \in 2^A \setminus \mathbb{S}_{E^-}, w_e = 0\}$ s.t. $S = S_{e''}$ (best solution for synthesizing set of arguments not among positive examples). One can compute e'' in polynomial time. Computing weights for all elements in E^+ and e'' yields an optimal solution AF. □

5 CONSTRAINT-BASED SYNTHESIS OF AFs

We continue by presenting MaxSAT encodings of AF synthesis. For background on MaxSAT, recall that for a Boolean variable x, there are two literals, x and $\neg x$. A clause is a disjunction ($\vee$) of literals. A truth assignment τ is a function from variables to true (1) and false (0). Satisfaction is defined as usual. A weighted partial MaxSAT (or simply MaxSAT) instance consists of hard clauses φ_h, soft clauses φ_s, and a weight function w associating to each soft clause $C \in \varphi_s$ a positive weight $w(C)$. An assignment τ is a solution to a MaxSAT instance $(\varphi_h, \varphi_s, w)$ if τ satisfies φ_h. The cost of τ, $c(\tau)$, is the sum of weights of the soft clauses not satisfied by τ. A solution τ to MaxSAT instance φ is optimal if $c(\tau) \leq c(\tau')$ for any solution τ' to φ.

Let $P = (A, E^+, E^-, \sigma)$ be an AF synthesis instance with $A = \{a_1, \ldots, a_n\}$ the set of arguments, E^+ the set of positive examples, E^- the set of negative examples, and $\sigma \in \{cf, adm, stb, com\}$ a semantics. In order to synthesize an optimal solution AF $F = (A, R)$ for P, we declare propositional variables $\text{Ext}_\sigma^{S_e}$ for each $e \in E^+ \cup E^-$, and $r_{a,b}$ for each $a, b \in A$. Now $\tau(\text{Ext}_\sigma^{S_e}) = 1$ indicates $S_e \in \sigma(F)$, and $\tau(r_{a,b}) = 1$ indicates $(a,b) \in R$. The hard clauses are for each $e \in E^+ \cup E^-$ equivalences of the form

$$\text{Ext}_\sigma^{S_e} \leftrightarrow \varphi_\sigma(S_e),$$

where $\varphi_\sigma(S_e)$ encodes the fact that S_e is a σ-extension. In other words, for conflict-free sets we have

$$\varphi_{cf}(S_e) = \bigwedge_{a,b \in S_e} \neg r_{a,b},$$

stating that no attacks should occur between arguments in the example. Admissible sets are encoded as

$$\varphi_{adm}(S_e) = \varphi_{cf}(S_e) \wedge \bigwedge_{a \in S_e} \bigwedge_{b \in A \setminus S_e} \left(r_{b,a} \rightarrow \bigvee_{c \in S_e} r_{c,b} \right),$$

that is, the extension is conflict-free and every argument in the set is defended. Likewise, if an example is a stable extension, it is conflict-free and its range is the whole set of arguments, encoded as

$$\varphi_{stb}(S_e) = \varphi_{cf}(S_e) \wedge \bigwedge_{a \in A \setminus S_e} \left(\bigvee_{b \in S_e} r_{b,a} \right).$$

Finally, we note that some further semantics can be covered in a similar fashion; for instance, an NP encoding for complete semantics is

$$\varphi_{com}(S_e) = \varphi_{adm}(S_e) \wedge \bigwedge_{a \in A \setminus S_e} \left(\bigvee_{b \in A} \left(r_{b,a} \wedge \bigwedge_{c \in S_e} \neg r_{c,b} \right) \right),$$

ensuring that every argument that is defended is included.

The soft clauses, on the other hand, encode the objective function of AF synthesis under minimization. For each $e \in E^+$, we have a soft clause $\text{Ext}_\sigma^{S_e}$, and for each $e \in E^-$, a soft clause $\neg\text{Ext}_\sigma^{S_e}$, with corresponding weights. An optimal solution to an AF synthesis instance is directly extracted from an optimal solution τ to the MaxSAT instance by including (a,b) to the attack structure iff $\tau(r_{a,b}) = 1$.

MaxSAT also allows for declaring additional constraints on the solution AFs of interest by encoding such constraints as hard (or soft) clauses. For instance, any particular attack (a,b) can be fixed by including the hard clause $(r_{a,b})$. Furthermore, one can for instance also synthesize an AF with the minimum number of attacks satisfying the maximum number of examples by adding soft clauses which state that the secondary preference is minimizing the number of attacks in the style of multi-level Boolean optimization [1]; in this case, in order to still guarantee that the primary objective of satisfying the maximum number of examples is met, the weights of the examples can be adjusted to be larger than the sum of the weights imposed on adding individual attacks to the solution AFs.

6 EXPERIMENTS

We implemented our MaxSAT encodings; the resulting system and benchmarks are available at `http://www.cs.helsinki.fi/group/coreo/afsynth/`. Here we present results from a first empirical evaluation of the scalability of the approach. The experiments were run on 2.83-GHz Intel Xeon E5440 quad-core machines with 32-GB memory and Debian GNU/Linux 8 using a per-instance timeout of 900 seconds. For the experiments, we used the state-of-the-art MaxSAT solver MSCG [21].

We used two different approaches to construct AF synthesis instances. The first set of benchmarks was generated based on the benchmark AFs used in the ICCMA'15 competition [26] as follows. We selected all AFs among the benchmarks that have at least five stable extensions. The number of arguments in these 17 AFs ranges from 141 to 964. For each AF, we picked uniformly at random 5 positive examples from the set of extensions. To obtain negative examples, we selected $10, 20, \dots, 150$ subsets of $\bigcup \mathbb{S}_{E^+}$ uniformly at random, using $p_{\text{arg}} = \frac{\sum_{e \in E^+} |S_e|/|E^+|}{|\bigcup \mathbb{S}_{E^+}|}$ as the probability of including an argument in a negative example. For intuition, this choice of p_{arg} makes the sizes of positive and negative examples approximately the same. Letting $A = \bigcup \mathbb{S}_{E^+}$ resulted in instances containing 54 to 370 arguments. Further, we introduced noise by swapping each of the initial positive and negative examples with a probability $p_{\text{noise}} \in \{0, 0.25, 0.5\}$. Weights were associated to each example by picking uniformly at random integers from the interval $[1, 10]$.

The second set of benchmarks was generated using the following random model. We picked $5, 10, \dots, 80$ positive examples from a set of 100 arguments uniformly at random with probability $p_{\text{arg}}^+ = 0.25$. Then $|E^-| = 20, 40, \dots, 200$ negative examples were sampled from the set $A = \bigcup \mathbb{S}_{E^+}$, and each argument was included with probability $p_{\text{arg}}^- = \frac{\sum_{e \in E^+} |S_e|/|E^+|}{|\bigcup \mathbb{S}_{E^+}|}$. Again, each example was assigned as weight a random integer from the interval $[1, 10]$. For each choice of parameters, this procedure was repeated 10 times to obtain a representative set of benchmarks.

A summary of the results for the admissible and stable semantics is shown in Figure 2. We exclude results for ICCMA instances under admissible, as these turned out to be very easy for our approach until running out of memory due to the increasing size of the encoding. For the ICCMA instances under the stable semantics (Figure 2 left), almost every instance can be solved within the timeout limit for up to 100 examples, with a median running time of only ≈ 10 seconds at 100 examples. Increasing the noise probability clearly increases hardness; we hypothesize this to be due to the fact that by increasing noise we are increasing the number of positive examples. On the random instances the number of negative examples under the admissible semantics (Figure 2 right) clearly correlates with runtimes. Under the stable semantics (Figure 2 middle), the number of negative examples does not appear to have a noticeable effect on the runtimes. This is inline with our complexity analysis (recall Section 4), as under the stable semantics AF synthesis remains NP-complete even without any negative examples, unlike under admissible.

7 VARIANTS AND EXTENSIONS

We discuss further variants of AF synthesis: synthesis from multiple sources, synthesis under multiple semantics, and synthesis from symbolically represented examples.

Multiple Sources. The problem of AF synthesis is in a natural way applicable when the examples originate from multiple sources, e.g., collections of extensions of several source AFs, and resulting in an AF that optimally solves the task of synthesizing the union of semantical collection or examples of the different sources. This use of the AF synthesis problem shares resemblance with aggregation or merging of multiple AFs studied in [12, 11].

Multiple Semantics. So far for each AF synthesis problem we required that all examples are given w.r.t. a specific semantics. A natural generalization is to let each example individually be linked to a semantics. Formally, an example $e = (S, w, \sigma)$ is then a triple with a semantics σ, denoted by σ_e. The cost of an AF F is given by

$$\sum_{e \in E^+} w_e \cdot I(S_e \notin \sigma_e(F)) + \sum_{e \in E^-} w_e \cdot I(S_e \in \sigma_e(F)).$$

Example 5. *Consider $E^+ = \{(\{a,c\}, 1, cf), (\{b,c\}, 1, stb)\}$ and $E^- = \{(\{a\}, 1, adm)\}$. This defines a unique 0-cost AF $F =$*

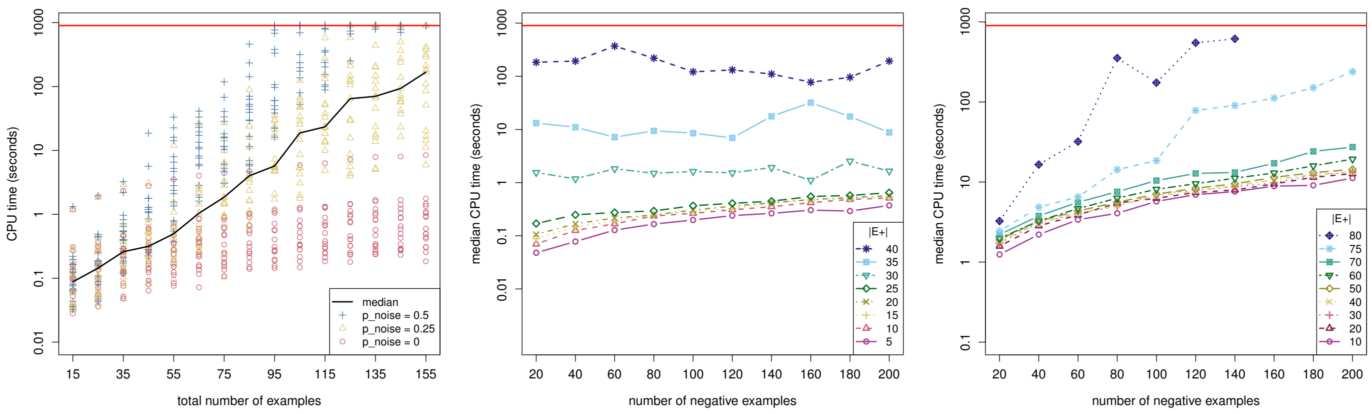

Figure 2. ICCMA instances for stable semantics (left); random instances for stable (middle) and admissible (right).

(A, R) *with* $A = \{a, b, c\}$ *for the AF synthesis instance with multiple semantics* $P = (A, E^+, E^-)$ *by* $R = \{(b, a)\}$.

The corresponding decision problem, i.e., for a given AF synthesis problem with multiple semantics $P = (A, E^+, E^-)$, is there an AF F with $cost(P, F) \leq k$ for an integer $k \geq 0$, does not exhibit higher computational complexity among the conflict-free, admissible, and stable semantics.

Corollary 11. *AF synthesis with multiple semantics among the conflict-free, admissible, and stable semantics is* NP*-complete.*

For solving AF synthesis with multiple semantics we can make use of our encodings of Section 5. In particular, we can conjoin the corresponding formulas for the different semantics and sharing the variables for attacks.

Symbolic Representation of Examples. For a set A of arguments, there can be in general up to $2^{|A|}$ positive or negative examples. This exponentiality in the input can be restrictive for large number of examples. Following ideas from [15], we note that, instead of explicit representation, examples could also be represented symbolically by encoding them succinctly in Boolean logic. This gives rise to the problem variant of AF synthesis with symbolic representation, with instances of the form $P = (A, \phi^+, \phi^-, \sigma)$, where ϕ^+ and ϕ^- are Boolean formulas. Let $Mod(\phi)$ be the set of models (satisfying assignments) of a Boolean formula ϕ, represented as sets themselves (variables assigned to true). For a given AF F, its associated cost $cost(P, F)$ is

$$\sum_{m \in Mod(\phi^+)} I(m \notin \sigma(F)) + \sum_{m \in Mod(\phi^-)} I(m \in \sigma(F)),$$

that is, unit weight is applied when a model of ϕ^+ is not a σ-extension of F and when a model of ϕ^- is a σ-extension of F.

Lemma 12. *Let* ϕ *be a Boolean formula,* A *the vocabulary of* ϕ*,* F *an AF, and* σ *a semantics. It holds that* $|Mod(\phi)| = cost(P, F)$ *for* $P = (A, \phi, \phi, \sigma)$.

Proof. Any AF F satisfies exactly $|Mod(\phi)|$ examples encoded in the formulas, since if $m \in Mod(\phi)$, then either $m \in \sigma(F)$ or $m \notin \sigma(F)$ (each implies unit weight). If $m \notin Mod(\phi)$, then both $m \in \sigma(F)$ and $m \notin \sigma(F)$ imply no weight. □

Based on this lemma, determining the cost of a given AF for an AF synthesis instance with symbolic representation is presumably very complex. In particular, we show #P-hardness, #P being the class of counting problems where the task is to count the number of accepting paths of a given non-deterministic polynomial-time Turing machine (see [27, 28]). As a prominent example, counting the number of models of a Boolean formula is #P-complete.

Corollary 13. *Counting the number of examples from a given AF synthesis instance with symbolic representation that are not satisfied by a given AF is #P-complete under the conflict-free, admissible, complete, and stable semantics.*

8 RELATED WORK

Before conclusions, we briefly discuss some of the most related work to ours within and beyond argumentation.

As we generalize the notion of realizability, the most closely related work to ours are recent articles focusing on realizability of AFs, and most recently, of abstract dialectical frameworks (ADFs) [8, 23, 19]. The central question studied in these works, as initiated in [15], is whether a given set of sets (of arguments) $\mathbb{S}$ can be realized by an AF, i.e., whether an AF with $\sigma(F) = \mathbb{S}$ for a semantics σ exists. In [3, 20] realizability was studied under the restriction that the set of arguments of the constructed AF has to match exactly the set of arguments occurring in the input, i.e., is in $\mathbb{S}$. In [16] the authors give a construction for an AF using additional arguments in the three-valued labeling setting under the preferred and semi-stable semantics. We study the problem of synthesizing an AF that optimally matches a given set of examples semantically, even when an exact realization (a 0-cost solution) is not possible. Also, we analyze the complexity of AF synthesis, showing that, in contrast to polynomial-time results for checking realizability [15], AF synthesis is in general NP-complete. To our best knowledge, no previous systems for solving realizability have been empirically evaluated. Most recently, in [23, 19] a declarative encoding in answer set programming (ASP) for realizability was presented but not empirically evaluated. Our MaxSAT-based implementation for the AF synthesis problem also covers realizability.

In related work that incorporates AF construction from examples, [22] formally studies a logical characterization of inductive concept learning and AF learning in a multi-agent setting. In contrast to our work, they induce a rule-based theory and construct an AF based on conflicting rules. Very recently, [24] study probabilistic AF [17, 18, 25] learning with non-exact methods; we tackle the exact optimization problem of AF synthesis.

Finally, going beyond argumentation, there is a long line of research of constructing, inducing, or learning logical structures from examples, from [29] to, e.g, work for constraint acquisition [7] as well as inductive logic programming [10, 6].

9 CONCLUSIONS

We proposed AF synthesis as a generalization of the important problem of realizability in abstract argumentation, relaxing in a natural way the stringent requirements for realizability to accommodate incomplete and noisy information. From the theoretical perspective, we related AF synthesis to realizability, and analyzed the complexity of AF synthesis both in the general case and in restricted settings. Motivated by the NP-completeness of AF synthesis in general under three key AF semantics, we proposed Boolean optimization based algorithmic solutions for the problem, and empirically studied this first approach to AF synthesis using a state-of-the-art MaxSAT solver on different types of AF synthesis instances. In terms of further work, we hope to establish the computational complexity of AF synthesis under further central AF semantics, and thereafter extend the MaxSAT-based approach to cover additional semantics.

ACKNOWLEDGEMENTS

Work funded by Academy of Finland, grants 251170 COIN, 276412, and 284591; and Research Funds of the University of Helsinki.

A FORMAL PROOFS

We provide formal proofs for the results presented in Sections 3 and 4. We start by restating definitions from [15].

Definition 4. *([15, Definitions 6, 7, and 8]) Let A be a set of arguments and $\mathbb{S} \subseteq 2^A$ a set of sets of arguments. Set $\mathbb{S}$ is*

- *incomparable if $S \not\subseteq S'$ holds for all $S, S' \in \mathbb{S}$ with $S \neq S'$;*
- *tight if for all $S \in \mathbb{S}$ and all $a \in A$ it holds that $S \cup \{a\} \notin \mathbb{S}$ implies that there exists a $b \in S$ s.t. $\{a, b\} \not\subseteq S'$ for all $S' \in \mathbb{S}$;*
- *conflict-sensitive if for each $S, S' \in \mathbb{S}$ s.t. $S \cup S' \notin \mathbb{S}$ it holds that $\exists a, b \in S \cup S'$ s.t. $\{a, b\} \not\subseteq S''$ for all $S'' \in \mathbb{S}$; and*
- *downward closed if $\mathbb{S} = \{S' \mid S \in \mathbb{S}, S' \subseteq S\}$.*

Proof of Lemma 1. Assume that $\mathbb{S} \subseteq cf(F)$ holds. Let $S \in \mathit{ImpliedCF}(\mathbb{S})$. By definition it follows that for each $a, b \in S$ we have $\exists S' \in \mathbb{S}$ with $\{a, b\} \subseteq S'$. If $\mathbb{S} \subseteq cf(F)$ then $S' \in cf(F)$ and thus $\{a, b\} \in cf(F)$. This implies that $S \in cf(F)$ (each pair in S is conflict-free). □

For proving Proposition 2 we use the following auxiliary lemma.

Lemma 14. *Let $\mathbb{S} \subseteq 2^A$ for a set A. It holds that $\mathit{ImpliedCF}(\mathbb{S})$ (i) contains $\emptyset$, (ii) is downward closed, (iii) is tight, and (iv) $\mathbb{S} \subseteq \mathit{ImpliedCF}(\mathbb{S})$.*

Proof. From the definition it directly follows that $\emptyset$ is contained in $\mathit{ImpliedCF}(\mathbb{S})$ for any $\mathbb{S}$. Let $S \in \mathit{ImpliedCF}(\mathbb{S})$. Then for all $a, b \in S$ it holds that $\exists S' \in \mathit{ImpliedCF}(\mathbb{S})$ s.t. $\{a, b\} \subseteq S'$. It follows that for any $S'' \subseteq S$ it holds that $S'' \in \mathit{ImpliedCF}(\mathbb{S})$. To show (iii), suppose that the set is not tight, i.e., $\exists S \in \mathit{ImpliedCF}(\mathbb{S})$ and $a \in A$ s.t. $S \cup \{a\} \notin \mathit{ImpliedCF}(\mathbb{S})$ and for all $b \in S$ there exists an $S' \in \mathit{ImpliedCF}(\mathbb{S})$ s.t. $\{a, b\} \subseteq S'$. This implies that for all $x, y \in S \cup \{a\}$ we have $\exists S'' \in \mathit{ImpliedCF}(\mathbb{S})$ s.t. $\{x, y\} \subseteq S''$ and thus $S \cup \{a\} \in \mathit{ImpliedCF}(\mathbb{S})$ which contradicts the assumption that $\mathit{ImpliedCF}(\mathbb{S})$ is not tight. Finally, if $S \in \mathbb{S}$ then it follows that $S \in \mathit{ImpliedCF}(\mathbb{S})$ (iv). □

Proof of Proposition 2. For the "only-if" direction, assume AF F is an optimal solution to P of cost 0, i.e., $cost(P, F) = 0$. It follows that $\mathbb{S}_{E+} \subseteq cf(F)$. By Lemma 1 we have $\mathit{ImpliedCF}(\mathbb{S}_{E+}) \subseteq cf(F)$. Thus $\mathit{ImpliedCF}(\mathbb{S}_{E+}) \cap \mathbb{S}_{E-} = \emptyset$, since $cf(F) \cap \mathbb{S}_{E-} = \emptyset$. For the "if" direction, assume $\mathit{ImpliedCF}(\mathbb{S}_{E+}) \cap \mathbb{S}_{E-} = \emptyset$. By Lemma 14 it follows that $\mathit{ImpliedCF}(\mathbb{S}_{E+})$ is tight, downward closed, and contains $\emptyset$. Due to [15, Proposition 5] it immediately follows that there exists an AF $F = (A', R')$ s.t. $cf(F) = \mathit{ImpliedCF}(\mathbb{S}_{E+})$. Since $\mathit{ImpliedCF}(\mathbb{S}_{E+}) \cap \mathbb{S}_{E-} = \emptyset$, it follows that $cost(P, F) = 0$. In [15, Proposition 5] the set A' is specified as all arguments occurring in $\mathit{ImpliedCF}(\mathbb{S}_{E+})$. If A contains more arguments, then we construct $F = (A, R)$ by extending R' with self-attacks for each $A \setminus A'$. □

Proof of Lemma 3. Assume $\mathbb{S} \subseteq adm(F)$ holds and let $S \in \mathit{ImpliedADM}(\mathbb{S})$. Then $S \in \mathit{ImpliedCF}(\mathbb{S})$ and thus $S \in cf(F)$ (Lemma 1). Finally, every union of admissible sets which is conflict-free is again an admissible set (see [9, Lemma 1]). □

For proving Proposition 4 we utilize the following lemma.

Lemma 15. *Let $\mathbb{S} \subseteq 2^A$ for a set A. It holds that $\mathit{ImpliedADM}(\mathbb{S})$ (i) contains $\emptyset$ and (ii) is conflict-sensitive.*

Proof. Claim (i) follows from definition. Suppose (ii) does not hold, i.e., there exist $S, S' \in \mathit{ImpliedADM}(\mathbb{S})$ s.t. $S \cup S' \notin \mathit{ImpliedADM}(\mathbb{S})$ and for all $a, b \in S \cup S'$ there exists an $S'' \in \mathit{ImpliedADM}(\mathbb{S})$ with $\{a, b\} \subseteq S''$. It follows that $S \cup S' \in \mathit{ImpliedCF}(\mathbb{S})$ and $S \cup S' \in \mathit{ImpliedADM}(\mathbb{S})$. This contradicts the assumption that $\mathit{ImpliedADM}(\mathbb{S})$ is not conflict-sensitive. □

Proof of Proposition 4. For the first claim, assume that there exists an AF F with $cost(P, F) = 0$. Then $\mathbb{S}_{E+} \subseteq adm(F)$ and thus $\mathit{ImpliedADM}(\mathbb{S}_{E+}) \subseteq adm(F)$, due to Lemma 3. For the second claim, assume $\mathit{ImpliedADM}(\mathbb{S}_{E+}) \cap \mathbb{S}_{E-} = \emptyset$ and condition 2 holds. By Lemma 15, $\mathit{ImpliedADM}(\mathbb{S}_{E+})$ is conflict-sensitive and contains $\emptyset$. By [15, Proposition 8], there exists an AF $F' = (A', R')$ s.t. $A' \subseteq A$ and $adm(F') = \mathit{ImpliedADM}(\mathbb{S}_{E+})$. Define $F = (A, R)$ by extending R' to have self-attacks for each argument in $A \setminus A'$. It follows that $adm(F) = \mathit{ImpliedADM}(\mathbb{S}_{E+})$. Assuming conditions 1-2, we have $cost(P, F) = 0$. □

Proof of Proposition 5. The claims of the proposition follow directly for the special case with $E^+ = \emptyset$. We proceed with the case that E^+ is non-empty. For the first claim, assume that F is a 0-cost solution to P. Conditions 1, 3, and 4 follow immediately. For condition 2, since $\mathbb{S}_{E+} \subseteq stb(F)$ it follows that $\mathbb{S}_{E+} \subseteq cf(F)$ and thus $\mathit{ImpliedCF}(\mathbb{S}_{E+}) \subseteq cf(F)$. Supposing condition 2 does not hold directly violates that $\mathbb{S}_{E+} \subseteq stb(F)$ (stable extensions are subset-maximal conflict-free sets). For the second claim, assume that conditions 1-5 hold. Then $\mathbb{S}_{E+}$ is a subset of the $\subseteq$-maximal elements of $\mathit{ImpliedCF}(\mathbb{S}_{E+})$, since $\mathbb{S}_{E+} \subseteq \mathit{ImpliedCF}(\mathbb{S}_{E+})$ (Lemma 14), and by assumption of condition 2. By [15, Lemma 2] it follows that $\mathbb{S}_{E+}$ is tight. Further, by [15, Proposition 7] and conditions 1-5, it follows that there exists an AF $F' = (A', R')$ with $A' \subseteq A$ s.t. $stb(F') = \mathbb{S}_{E+}$. Construct $F = (A, R)$ by extending R' to contain self-attacks for each argument in $A \setminus A'$ and attacks from each argument in $\mathbb{S}_{E+}$ to each $A \setminus A'$. □

We continue with proofs for the special cases of empty E^+ or E^-.

Proof of Proposition 6. If the first condition is met, AF $F = (A, R)$ with $R = (A \times A)$ satisfies $cf(F) = adm(F) = \{\emptyset\}$ and $stb(F) =$

$\emptyset$. If the second condition is met, AF $F' = (A, \emptyset)$ satisfies $cf(F') = adm(F') = 2^A$. □

We move on to proofs of complexity results.

Proof of Proposition 7. For an AF synthesis instance $P = (A, E^+, E^-, cf)$ membership in NP follows from a guess of an AF $F = (A, R)$, since $cost(P, F)$ can be computed in polynomial time.

For hardness, we provide a reduction from the satisfiability problem of conjunctive normal form (CNF) Boolean formulas. Let ϕ be a propositional formula in 3-CNF over variables $X = \{x_1, \dots, x_n\}$, $|X| = n$ and set of clauses C. Let $b = n + 1$.

$$E^+ = \{(\{x_i, x_i^T\}, 1) \mid x_i \in X\} \cup \quad (1)$$
$$\{(\{x_i, x_i^F\}, 1) \mid x_i \in X\} \cup \quad (2)$$
$$\{(\{x_i^T, x_i^F\}, b) \mid x_i \in X\} \cup \quad (3)$$
$$\{(\{y, z\}, b) \mid x_i, x_j \in X, i \neq j, y \in \{x_i, x_i^T, x_i^F\}, z \in \{x_j, x_j^T, x_j^F\}\} \quad (4)$$
$$E^- = \{(\{x_i, x_i^T, x_i^F\}, b) \mid x_i \in X\} \cup \quad (5)$$
$$\{(\{x_i, x_i^T \mid x_i \in c\} \cup \{x_i, x_i^F \mid \neg x_i \in c\}, b) \mid c \in C\} \quad (6)$$

Let $P = (A, E^+, E^-, cf)$ be the constructed instance for AF synthesis with bound $k = n$. Intuitively, one cannot satisfy all examples of forms (1), (2), (3), and (5) simultaneously, and due to the chosen weights and cost limit, one has to violate either (1) or (2) for a given $x_i \in X$, thus "choosing" a truth assignment over X (true iff an attack between x_i and x_i^F is synthesized). We now claim that there exists an AF $F = (A, R)$ with $cost(P, F) \leq n$ iff ϕ is satisfiable.

"Only-if" direction: assume that $F = (A, R)$ has $cost(P, F) \leq n$. Then all examples with weight $n + 1$ are satisfied by F. It is immediate that for each $x_i \in X$ we have either $\{x_i, x_i^T\} \in cf(F)$ or $\{x_i, x_i^F\} \in cf(F)$ but not both (if both would be conflict-free then together with $\{x_i^T, x_i^F\}$ being conflict-free in F implies that $\{x_i, x_i^T, x_i^F\}$ if conflict-free which contradicts that cost of F is lower than $n + 1$ (5); if none of the sets with weight 1 are conflict-free for $x_i \in X$, then overall cost would be over n as well). This straightforwardly defines a truth assignment $\tau(x_i) = 0$ iff $\{x_i, x_i^T\} \in cf(F)$ and 1 otherwise. Suppose τ does not satisfy ϕ. Then there exists a $c \in C$ s.t. $\tau \not\models c$ and τ does not satisfy any literal l in c.

$$\begin{aligned}
&\tau(l) = 0, \forall l \in c \\
&\text{iff } \forall l \in c \\
&\qquad l = x_i \text{ implies } \tau(x_i) = 0 \text{ and} \\
&\qquad l = \neg x_i \text{ implies } \tau(x_i) = 1 \\
&\text{iff } \forall l \in c \\
&\qquad l = x_i \text{ implies } \{x_i, x_i^T\} \in cf(F) \text{ and} \\
&\qquad l = \neg x_i \text{ implies } \{x_i, x_i^F\} \in cf(F) \\
&\text{iff } \{x_i, x_i^T \mid x_i \in c\} \cup \{x_i, x_i^F \mid \neg x_i \in c\} \in cf(F)\ (*) \\
&\text{only-if } cost(P, F) \geq n + 1
\end{aligned}$$

Conclusion $(*)$ follows from (4): for each $x_i, x_j \in X$ with $x_i \neq x_j$ no attacks are in between sets $\{x_i, x_i^T, x_i^F\}$ and $\{x_j, x_j^T, x_j^F\}$. "If" direction: assume ϕ is satisfiable. Construct AF $F = (A, R)$ with $R = \{(x_i, x_i^T) \mid \tau(x) = 0\} \cup \{(x_i, x_i^F) \mid \tau(x) = 1\}$. It is immediate that F satisfies all non-unit weighted examples except for (6), which follows from similar consideration as in the only-if direction. Finally, cost of F is n. □

Proof sketch of Proposition 8. NP-completeness for admissible semantics follows the same reasoning as proof of Proposition 7. For the "only-if" direction, just note that the same unit-weighted examples are mutually exclusive for a solution AF as in the conflict-free case. For "if" direction, the constructed AF has mutual attacks instead of uni-directional ones (conflict-free sets are then admissible). □

Proof sketch of Proposition 9. Membership follows from a non-deterministic guess of an AF F and checking each example individually whether it is satisfied (which can be done in polynomial time).

For hardness, we provide a reduction from the Boolean satisfiability problem. Let ϕ be a Boolean formula in CNF, with vocabulary X, with $|X| = n$, and set of clauses C. Let $b = n + 1$.

$$A = X \cup \{x^T, x^F, d_x \mid x \in X\} \cup \{d'_c, d''_c \mid c \in C\} \cup \{d\} \quad (7)$$
$$E^+ = \{(\{d'_c\} \cup \{x^T \mid x \in c\} \cup \{x^F \mid \neg x \in c\}, b) \mid c \in C\} \cup \quad (8)$$
$$\{(\{d'_c, d''_c, d\}, b) \mid c \in C\} \cup \quad (9)$$
$$\{(\{x^T, x^F, d_x\}, b) \mid x \in X\} \cup \quad (10)$$
$$\{(\{x^T, d_x, d\}, 1) \mid x \in X\} \cup \quad (11)$$
$$\{(\{x^F, d_x, d\}, 1) \mid x \in X\} \quad (12)$$

Let $P = (A, E^+, \emptyset, stb)$ and bound $k = n$. We claim that ϕ satisfiable iff there exists an AF F s.t. $cost(P, F) \leq n$.

Assume such an AF F exists. It is immediate that all examples with weight $n + 1$ are satisfied by F. For each $x \in X$ it holds that exactly one example of $\{(\{x^T, d_x, x\}, 1), (\{x^F, d_x, x\}, 1)\}$ is satisfied. If none of them is satisfied the cost of F would be higher than n. If both are satisfied, then there is no attack between x^T, x^F, d_x, and d, thus $\{x^T, x^F, d_x\}$ cannot be stable (does not attack d). This defines a truth assignment $\tau(x) = 0$ iff $(\{x^T, d_x, d\}, 1)$ is satisfied by F. We claim that $\tau \models \phi$. Suppose the contrary, then $\exists c \in C$ with $\tau \not\models c$ and all literals in c are not satisfied by τ. Consider the set $\{d'_c\} \cup \{x^T \mid x \in c\} \cup \{x^F \mid \neg x \in c\}$, which must be stable in F by assumption. By construction of τ and example (9), it follows that no argument in that set attacks d, thus it cannot be stable, which contradicts the assumption that τ does not satisfy ϕ.

Assume that ϕ is satisfiable. Construct AF $F = (A, R)$ with

$$\begin{aligned}
R = &\{(d''_c, d_x), (d_x, d''_c) \mid c \in C, x \in X\} \cup \\
&\{(d''_c, x^T), (d''_c, x^F), (x^T, d''_c), (x^F, d''_c) \mid c \in C, x \in X\} \cup \\
&\{(a, b), (b, a) \mid a \in \{d'_c, d''_c\}, b \in \{d'_{c'}, d''_{c'}\}, c, c' \in C, c \neq c'\} \cup \\
&\{(d_x, d_y), (d_x, z), (z, d_x) \mid x, y \in X, x \neq y, z \in \{y^T, y^F\}\} \cup \\
&\{(d_x, d'_c), (d'_c, d_x) \mid c \in C, x \in X\} \cup \\
&\{(d'_c, x^T), (x^T, d'_c) \mid c \in C, x \in X, x \notin c\} \cup \\
&\{(d'_c, x^F), (x^F, d'_c) \mid c \in C, x \in X, \neg x \notin c\} \cup \\
&\{(x^T, d), (d, x^T) \mid \tau(x) = 1\} \cup \\
&\{(x^F, d), (d, x^F) \mid \tau(x) = 0\}.
\end{aligned}$$

Briefly put, this involved construction adds mutual attacks between arguments to ensure that all the examples (9) and (10) are stable. A mutual attack is added between x^T and d (x^F and d) based on the truth assignment τ, violating one of the unit weighted examples. Finally, examples of form (8) are satisfied, since one of the arguments in these sets (except d'_c) attacks d due to assumption that τ satisfies ϕ. Intuitively, the examples encoding the clauses (8) are satisfied since one of the arguments corresponding to the chosen truth assignment that satisfies one of the literals attacks d. □

REFERENCES

[1] Josep Argelich, Inês Lynce, and João P. Marques-Silva, 'On solving Boolean multilevel optimization problems', in *Proc. IJCAI*, ed., Craig Boutilier, pp. 393–398. AAAI Press, (2009).

[2] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin, 'An introduction to argumentation semantics', *Knowledge Engineering Review*, **26**(4), 365–410, (2011).

[3] Ringo Baumann, Wolfgang Dvořák, Thomas Linsbichler, Hannes Strass, and Stefan Woltran, 'Compact argumentation frameworks', in *Proc. ECAI*, eds., Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pp. 69–74. IOS Press, (2014).

[4] Ringo Baumann and Hannes Strass, 'On the maximal and average numbers of stable extensions', in *Proc. TAFA*, eds., Elizabeth Black, Sanjay Modgil, and Nir Oren, volume 8306 of *Lecture Notes in Computer Science*, pp. 111–126. Springer, (2013).

[5] T.J.M. Bench-Capon and Paul E. Dunne, 'Argumentation in artificial intelligence', *Artificial Intelligence*, **171**(10-15), 619–641, (2007).

[6] Francesco Bergadano and Daniele Gunetti, *Inductive logic programming - from machine learning to software engineering*, MIT Press, 1996.

[7] Christian Bessiere, Frédéric Koriche, Nadjib Lazaar, and Barry O'Sullivan, 'Constraint acquisition', *Artificial Intelligence*, (2015).

[8] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran, 'Abstract dialectical frameworks revisited', in *Proc. IJCAI*, ed., Francesca Rossi, pp. 803–809. AAAI Press / IJCAI, (2013).

[9] Martin Caminada, 'Comparing two unique extension semantics for formal argumentation: Ideal and eager', in *Proc. BNAIC*, eds., Mehdi Dastani and Edwin de Jong, pp. 81–87, (2007).

[10] Jesse Davis and Jan Ramon, eds. *Inductive Logic Programming - 24th International Conference, ILP 2014, Nancy, France, September 14-16, 2014, Revised Selected Papers*, volume 9046 of *Lecture Notes in Computer Science*. Springer, 2015.

[11] Jérôme Delobelle, Adrian Haret, Sébastien Konieczny, Jean-Guy Mailly, Julien Rossit, and Stefan Woltran, 'Merging of abstract argumentation frameworks', in *Proc. KR*, eds., Chitta Baral, James P. Delgrande, and Frank Wolter, pp. 33–42. AAAI Press, (2016).

[12] Jérôme Delobelle, Sébastien Konieczny, and Srdjan Vesic, 'On the aggregation of argumentation frameworks', in *Proc. IJCAI*, eds., Qiang Yang and Michael Wooldridge, pp. 2911–2917. AAAI Press, (2015).

[13] Martin Diller, Adrian Haret, Thomas Linsbichler, Stefan Rümmele, and Stefan Woltran, 'An extension-based approach to belief revision in abstract argumentation', in *Proc. IJCAI*, eds., Qiang Yang and Michael Wooldridge, pp. 2926–2932. AAAI Press, (2015).

[14] Phan Minh Dung, 'On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games', *Artificial Intelligence*, **77**(2), 321–358, (1995).

[15] Paul E. Dunne, Wolfgang Dvořák, Thomas Linsbichler, and Stefan Woltran, 'Characteristics of multiple viewpoints in abstract argumentation', *Artificial Intelligence*, **228**, 153–178, (2015).

[16] Sjur Kristoffer Dyrkolbotn, 'How to argue for anything: Enforcing arbitrary sets of labellings using AFs', in *Proc. KR*, eds., Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, pp. 626–629. AAAI Press, (2014).

[17] Anthony Hunter, 'Probabilistic qualification of attack in abstract argumentation', *International Journal of Approximate Reasoning*, **55**(2), 607–638, (2014).

[18] Hengfei Li, Nir Oren, and Timothy J. Norman, 'Probabilistic argumentation frameworks', in *Proc. TAFA*, eds., Sanjay Modgil, Nir Oren, and Francesca Toni, volume 7132 of *Lecture Notes in Computer Science*, pp. 1–16. Springer, (2011).

[19] Thomas Linsbichler, Jörg Pührer, and Hannes Strass, 'Characterizing realizability in abstract argumentation', in *Proc. NMR*, eds., Gabriele Kern-Isberner and Renata Wassermann, pp. 85–94, (2016).

[20] Thomas Linsbichler, Christof Spanring, and Stefan Woltran, 'The hidden power of abstract argumentation semantics', in *Proc. TAFA*, eds., Elizabeth Black, Sanjay Modgil, and Nir Oren, volume 9524 of *Lecture Notes in Computer Science*, pp. 146–162. Springer, (2015).

[21] Antonio Morgado, Alexey Ignatiev, and Joao Marques-Silva, 'MSCG: Robust core-guided MaxSAT solving', *Journal on Satisfiability, Boolean Modeling and Computation*, **9**, 129–134, (2015).

[22] Santiago Ontañón, Pilar Dellunde, Lluis Godo, and Enric Plaza, 'A defeasible reasoning model of inductive concept learning from examples and communication', *Artificial Intelligence*, **193**, 129–148, (2012).

[23] Jörg Pührer, 'Realizability of three-valued semantics for abstract dialectical frameworks', in *Proc. IJCAI*, eds., Qiang Yang and Michael Wooldridge, pp. 3171–3177. AAAI Press, (2015).

[24] Régis Riveret and Guido Governatori, 'On learning attacks in probabilistic abstract argumentation', in *Proc. AAMAS*, eds., Catholijn M. Jonker, Stacy Marsella, John Thangarajah, and Karl Tuyls, pp. 653–661. ACM, (2016).

[25] Régis Riveret, Dimitrios Korkinof, Moez Draief, and Jeremy V. Pitt, 'Probabilistic abstract argumentation: an investigation with boltzmann machines', *Argument & Computation*, **6**(2), 178–218, (2015).

[26] Matthias Thimm, Serena Villata, Federico Cerutti, Nir Oren, Hannes Strass, and Mauro Vallati, 'Summary report of the first international competition on computational models of argumentation', *AI Magazine*, **37**(1), 102–104, (2016).

[27] Leslie G. Valiant, 'The complexity of computing the permanent', *Theoretical Computer Science*, **8**, 189–201, (1979).

[28] Leslie G. Valiant, 'The complexity of enumeration and reliability problems', *SIAM Journal on Computing*, **8**(3), 410–421, (1979).

[29] Leslie G. Valiant, 'A theory of the learnable', *Communications of the ACM*, **27**(11), 1134–1142, (1984).