Complexity of Abstract Argumentation under a Claim-Centric View

Wolfgang Dvořák, Stefan Woltran
TU Wien
Vienna, Austria
{dvorak,woltran}@dbai.tuwien.ac.at

Abstract

Abstract argumentation frameworks have been introduced by Dung as part of an argumentation process, where arguments and conflicts are derived from a given knowledge base. It is solely this relation between arguments that is then used in order to identify acceptable sets of arguments. A final step concerns the acceptance status of particular statements by reviewing the actual contents of the acceptable arguments. Complexity analysis of abstract argumentation so far has neglected this final step and is concerned with argument names instead of their contents, i.e. their claims. As we outline in this paper, this is not only a slight deviation but can lead to different complexity results. We, therefore, give a comprehensive complexity analysis of abstract argumentation under a claim-centric view and analyse the four main decision problems under seven popular semantics. In addition, we also address the complexity of common sub-classes and introduce novel parameterisations – which exploit the nature of claims explicitly – along with fixed-parameter tractability results.

Introduction

Formal argumentation is a vibrant field within AI. On the one hand it provides genuine methods to model discourses or legal cases (Atkinson et al. 2017). On the other hand, it is closely related to – and gives an orthogonal view on – several formalisms from the AI domain, e.g. logic programming or nonmonotonic reasoning principles (Dung 1995; Wu, Caminada, and Gabbay 2009; Caminada et al. 2015). For both applications, a particular model is widely used which is known as *instantiation-based argumentation* (see e.g. (Gorogiannis and Hunter 2011)).

This instantiation process starts from a knowledge base (KB), which is potentially inconsistent. From KB, all possible arguments are constructed first. An argument typically contains a claim and a support which is a subset of KB and derives the claim. Next, the relationship between arguments is analysed. A standard model is to consider that argument α attacks argument β if the claim of α contradicts (parts of) the support of β. As soon as all arguments and attacks between arguments are given, one abstracts away from the contents of the arguments and it is only the remaining attack network that is evaluated, which is thus termed abstract argumentation framework (AF). Semantics for AFs then deliver a collection of sets of arguments which are understood as jointly acceptable; these sets are commonly referred to as extensions.

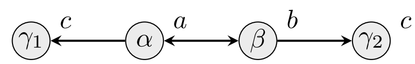


Figure 1: The AF F_P from Example 1. The claims of the arguments are given in the labels next to the argument.

Example 1 (Instantiating AFs from Logic Programs)**.** *Let $P = \{r_1 : a \leftarrow not\, b.;\quad r_2 : b \leftarrow not\, a.;\quad r_3 : c \leftarrow not\, a.;\quad r_4 : c \leftarrow not\, b.\}$ be a logic program (LP). The instantiation approach from (Caminada et al. 2015) yields an AF $F_P = (A, R)$ with arguments $A = \{\alpha, \beta, \gamma_1, \gamma_2\}$, where α represents rule r_1 and has claim a; β represents rule r_2 with claim b; γ_1 and γ_2 represent rules r_3 and r_4 respectively, and both have as their claim c. The attack relation R is constructed, such that an argument representing rule r attacks an argument representing rule r' if the head of r occurs negated in the rule body of r'. Hence, $R = \{(\alpha, \beta), (\beta, \alpha), (\alpha, \gamma_1), (\beta, \gamma_2)\}$; see Figure 1.*

Under this construction, stable model semantics of LPs corresponds to stable extensions of AFs (we omit technical details, they are not important for the sake of the argument; stable extensions of AFs will be formally introduced in the next section). In our example, the two stable models $S_1 = \{a, c\}$ and $S_2 = \{b, c\}$ of P are given via the two stable extensions $E_1 = \{\alpha, \gamma_2\}$ and $E_2 = \{\beta, \gamma_1\}$ of F_P. Note that the claims of E_1 yield S_1 and those of E_2 yield S_2. $\diamondsuit$

Having computed the extensions, the instantiation process is completed by re-interpreting these sets of arguments in terms of their claims. Typical are credulous and skeptical acceptance queries, which can be posed on argument *names* or their *claims*. For instance, for skeptical acceptance one might be interested whether a particular argument α is contained in all extensions (we will refer to this kind of reasoning as *argument-centric*). However, in the light of the above discussion the following question (which gives a *claim-centric* view) appears more appropriate

(SKEPT): is a particular claim c covered by all extensions, i.e. does every extension contain at least one argument with claim c?

Example 1 (continued). *With the extensions of F_P being $E_1 = \{\alpha, \gamma_2\}$ and $E_2 = \{\beta, \gamma_1\}$ of F_P, we note that no argument is skeptically accepted. However, c is a skeptical consequence of the program P. Hence, in order to check whether some claim is covered by each extension, we need to connect claims to their arguments, since argument acceptance alone is insufficient to decide this problem.* $\diamondsuit$

This subtle difference between skeptical reasoning on arguments and skeptical reasoning on claims has already been noticed by Prakken and Vreeswijk (2002)[Example 25] and is also discussed in the recent handbook-chapter on ASPIC (Modgil and Prakken 2018)[Def. 2.18 and below]. Indeed, computing the acceptance status of claims is important for instantiation-based argumentation systems, and understanding the complexity of this task is a key towards systems that perform sufficiently efficient in practical cases. In particular, identifying tractable cases for these problems is crucial. However, the existing literature on complexity analysis for (abstract) argumentation solely is concerned with reasoning over argument names (see e.g. (Dvořák and Dunne 2018)), while the claim-centric view seems neglected.

In this paper, we shall thus provide a comprehensive complexity analysis for decision problems on argumentation frameworks which are centred on claims rather than on arguments. We will study two scenarios: (1) AFs where claims are attached to arguments in an arbitrary way; (2) AFs where the assignments of claims to arguments satisfy a particular condition that reflects the assumption that an argument α attacks argument β if the claim of α contradicts (parts of) the support of β (like in the example above). Given that the attacks are constructed in that way, we have that arguments with the same claim attack the same arguments. We call such frameworks *well-formed*. Well-formed frameworks represent the most fundamental case for instantiation-based argumentation, while the more relaxed variant (1) applies to (more advanced) instantiations without such restrictions on the attack relation, which allows to take concepts like argument strength or preferences into account.

Main Contributions.

- We adapt four main decision problems studied in the literature to our proposed model and provide a complete complexity analysis for seven popular semantics. Our results demonstrate that switching from an argument-centric view to a claim-centric view can lead to higher complexity, in particular for the verification problem.
- We show that in the case of well-formed frameworks this divergence is less drastic, and it is only the skeptical acceptance of naive semantics that remains harder than in the argument-centric case.
- In addition, we also address the complexity of common sub-classes of frameworks when adapted to our settings and provide fixed-parameter tractability results. In particular, the concept of claims being attached to arguments gives rise to novel parameterisations which are inaccessible in the standard argument-centric view.

Preliminaries

Let us introduce argumentation frameworks (Dung 1995) and recall the semantics we study (for a comprehensive introduction, see (Baroni, Caminada, and Giacomin 2011)).

Definition 1. *An* argumentation framework (AF) *is a pair $F = (A, R)$ where A is a finite set of arguments and $R \subseteq A \times A$ is the attack relation. The pair $(a, b) \in R$ means that a attacks b, and we say that a set $S \subseteq A$ attacks (in F) an argument b if $(a, b) \in R$ for some $a \in S$. An argument $a \in A$ is* defended *(in F) by a set $S \subseteq A$ if each b with $(b, a) \in R$ is attacked by S in F.*

Semantics for argumentation frameworks are defined as 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 cf, $naive$, grd, stb, adm, com, and prf, which stand for conflict-free, naive, grounded, stable, admissible, complete, and preferred extensions, respectively.

Definition 2. *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 $(a, b) \in R$. $cf(F)$ denotes the collection of conflict-free sets of F. For a conflict-free set $S \in cf(F)$, it holds that*

- *$S \in naive(F)$, if there is no $T \in cf(F)$ with $T \supset S$;*
- *$S \in stb(F)$, if each $a \in A \setminus S$ is attacked by S in F;*
- *$S \in adm(F)$, if each $a \in S$ is defended by S in F;*
- *$S \in com(F)$, if $S \in adm(F)$ and each $a \in A$ defended by S in F is contained in S;*
- *$S \in grd(F)$, if $S \in com(F)$ and there is no $T \subset S$ such that $T \in com(F)$;*
- *$S \in prf(F)$, if $S \in adm(F)$ and there is no $T \supset S$ such that $T \in adm(F)$.*

Recall that for each AF F, $grd(F)$ yields a unique extension, the grounded extension of F; moreover, $stb(F) \subseteq naive(F)$ and $stb(F) \subseteq prf(F) \subseteq com(F) \subseteq adm(F)$.

The standard decision problems for an AF F w.r.t a semantics σ studied in the literature (see e.g. (Dvořák and Dunne 2018)) are: (a) $Cred_\sigma^{AF}$: Is an argument a contained in some extension $E \in \sigma(F)$? (b) $Skept_\sigma^{AF}$: Is an argument a contained in all extensions $E \in \sigma(F)$? (c) Ver_σ^{AF}: Is a given set E an extension, i.e. $E \in \sigma(F)$? and (d) $NEmpty_\sigma^{AF}$: Does there exist a non-empty extension $E \in \sigma(F)$?

Reasoning about Claims

To ease our claim-centric complexity analysis, we consider AFs augmented by claims as a distinguished concept. We simply associate a claim to each argument in an AF and redefine extensions in terms of the claims. This will allow us to rephrase in a natural way the standard decision problems for AFs under a claim-centric view.

Definition 3. *A* claim-augmented argumentation framework (CAF) *is a triple $(A, R, claim)$ where (A, R) is an AF and $claim : A \to \mathcal{C}$ assigns a claim to each argument of A; $\mathcal{C}$ is the set of possible claims.*

A CAF $(A, R, claim)$ is called well-formed *if, for any $a, b \in A$ with $claim(a) = claim(b)$, $\{c \mid (a, c) \in R\} = \{c \mid (b, c) \in R\}$, i.e. arguments with the same claim attack the same arguments.*

Note that different arguments can have the same claim. No further information about claims $\mathcal{C}$ will be available. In particular, we do not know whether different claims are in a certain equivalence relation to – or contradict – each other. However, the concept of well-formedness reflects certain effects of instantiating knowledge-bases into AFs, as discussed in the introduction.

The simplest way to decide questions like "is a certain claim covered by some/all extensions?" is to take standard semantics (as defined in the previous section) of the underlying AF, but interpret the extensions in terms of the claims of their arguments. In what follows, we extend the function $claim$ to sets, i.e. $claim(S) = \{claim(s) \mid s \in S\}$.

Definition 4. *For a semantics σ, we define its claim-based variant σ_c as follows. For any CAF $CF = (A, R, claim)$, $\sigma_c(CF) = \{claim(S) \mid S \in \sigma((A, R))\}$.*

We note that basic relations between different semantics carry over from standard AFs. In fact, for any CAF CF

$$stb_c(CF) \subseteq prf_c(CF) \subseteq com_c(CF) \subseteq adm_c(CF) \quad (1)$$

and $grd_c(CF)$ is unique and contained in $com_c(CF)$. Moreover, $stb_c(CF) \subseteq naive_c(CF)$.

General Complexity Results

The concept of CAFs now allows us to adopt typical computational problems to our needs (recall the (SKEPT) problem from the introduction) and to study the complexity of abstract argumentation under a claim-centric view. Given semantics σ, a CAF $CF = (A, R, claim)$, claim $c \in \mathcal{C}$, and claims $C \subseteq \mathcal{C}$ we consider the following decision problems.

- $Cred_\sigma^{CAF}$: Does $c \in S$ hold for at least one $S \in \sigma_c(CF)$? In other words, is c supported by at least one extension of (A, R), i.e. $c \in claim(E)$ for some $E \in \sigma((A, R))$?
- $Skept_\sigma^{CAF}$: Does $c \in S$ hold for all $S \in \sigma_c(CF)$? In other words, is c supported by all extensions of (A, R), i.e. $c \in claim(E)$ for each $E \in \sigma((A, R))$?
- Ver_σ^{CAF}: Does $C \in \sigma_c(CF)$ hold? In other words, is C the claim set of an extension of (A, R), i.e. $C = claim(E)$ for some $E \in \sigma((A, R))$?
- $NEmpty_\sigma^{CAF}$: Does $S \neq \emptyset$ hold for some $S \in \sigma_c(CF)$? In other words, is there an extension of (A, R) with non-empty claim set, i.e. $claim(E) \neq \emptyset$ for some $E \in \sigma((A, R))$?

We define these decision problems restricted to well-formed CAFs accordingly and denote them by $Cred_\sigma^{wf}$, $Skept_\sigma^{wf}$, Ver_σ^{wf}, and $NEmpty_\sigma^{wf}$.

The high level picture of the forthcoming results is that reasoning in CAFs is of the same complexity as in AFs (cf. (Dvořák and Dunne 2018)), except for naive semantics where skeptical reasoning goes up one level in the polynomial hierarchy (even for well-formed CAFs). Moreover, the verification problem is more expensive for CAFs than for AFs for most of the semantics, but this is not the case when restricted to well-formed CAFs.

Theorem 1. *The complexity results for CAFs as given in Table 1 hold (C-c denotes completeness for class C).*

Table 1: Complexity of CAFs. Results that deviate from the corresponding results for AFs are highlighted in bold-face.

σ	$Cred_\sigma^{CAF}$	$Skept_\sigma^{CAF}$	Ver_σ^{CAF}	$NEmpty_\sigma^{CAF}$
cf	in P	trivial	**NP**-c	in P
$naive$	in P	**coNP**-c	**NP**-c	in P
grd	P-c	P-c	P-c	in P
stb	NP-c	coNP-c	**NP**-c	NP-c
adm	NP-c	trivial	**NP**-c	NP-c
com	NP-c	P-c	**NP**-c	NP-c
prf	NP-c	Π_2^P-c	$\mathbf{\Sigma_2^P}$-c	NP-c

To start with, we discuss the results that are implications of the corresponding results for AFs: First, an extension has a non-empty claim set iff it is non empty. That is, $NEmpty_\sigma^{CAF}$ coincides with the corresponding problem for AFs and its complexity thus follows from the literature (Dvořák and Dunne 2018). Second, CAF problems generalise the corresponding problems for AFs (and are thus as least as hard); indeed, AF problems can be reduced to the corresponding CAF problems by assigning each argument a unique claim. Third, the non-trivial cases of $Cred_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ can be decided by small adaptations of the standard arguments. For instance, $Cred_{naive}^{CAF}$ holds iff there is one argument with the given claim that is not self-attacking; for $Cred_{stb}^{CAF}$ guess a set E of arguments with the given claim c contained in $claim(E)$ and check whether E is stable in the underlying AF, etc.

It remains to prove the complexity of the Ver_σ^{CAF} problems and $Skept_{naive}^{CAF}$, i.e. those which deviate from the complexity for AFs.

Proposition 1. *Ver_σ^{CAF} is NP-complete for $\sigma \in \{cf, naive, stb, adm, com\}$.*

Proof. NP-membership is by the following procedure. For $CF = (A, R, claim)$, a set C can be verified to be in $\sigma_c(CF)$ by guessing a set of arguments $E \subseteq A$ with $claim(E) = C$ and checking that E is a σ-extension of (A, R). The latter is in P by known results for AFs.

We next show that Ver_σ^{CAF} is NP-hard for $\sigma \in \{cf, naive, stb, adm, com, prf\}$. Consider the following reduction from 3-SAT where the formula φ is given as a set $Cl = \{cl_1, \ldots, cl_m\}$ of clauses over atoms X. We construct a CAF $CF = (A, R, claim)$ with the arguments given by the two sets $V = \{x_i \mid x \in X, x \in cl_i\}$ and $\bar{V} = \{\bar{x}_i \mid x \in X, \neg x \in cl_i\}$, i.e. $A = V \cup \bar{V}$. We set $R = \{(x_i, \bar{x}_j), (\bar{x}_j, x_i) \mid x_i \in V, \bar{x}_j \in \bar{V}\}$, $claim(x_i) = i$ and $claim(\bar{x}_i) = i$. See Figure 2 for an example to illus-

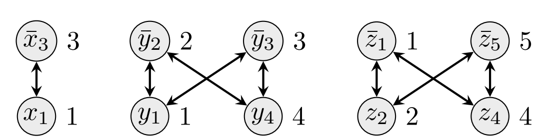


Figure 2: CAF from the proof of Prop. 1 for the formula φ with clauses $\{\{x, y, \neg z\}, \{\neg y, z\}, \{\neg x, \neg y\}, \{y, z\}, \{\neg z\}\}$.

trate the reduction. It can be checked that φ is satisfiable iff $\{1, \ldots, m\} \in \sigma_c(CF)$. $\square$

Proposition 2. *Ver_{prf}^{CAF} is Σ_2^{P}-complete.*

Proof. Membership is by the same procedure as in the proof of Proposition 1 and the fact that verifying whether a set of arguments is a preferred extension of an AF is in coNP.

For hardness, we use $Skept_{prf}^{AF}$ (i.e. skeptical acceptance for standard AFs) which is Π_2^{P}-hard, and reduce its complement to Ver_{prf}^{CAF}. Generally, for any semantics σ one can reduce co$Skept_{\sigma}^{AF}$ to Ver_{σ}^{CAF} as follows: consider an instance testing argument a for skeptical acceptance in (A, R) w.r.t. σ. We construct a CAF $CF = (A \cup \{i\}, R, claim)$ by adding an isolated argument i and defining the $claim$ function such that $claim(a) = c_1$ and $claim(b) = c_2$ for $b \in (A \setminus \{a\}) \cup \{i\}$. Then, a is not skeptically accepted in (A, R) w.r.t. σ iff $\{c_2\} \in \sigma_c(CF)$. $\square$

Proposition 3. *$Skept_{naive}^{CAF}$ is coNP-complete.*

Proof. The membership is by a classical guess and check algorithm. For hardness consider an instance of 3-SAT where the formula φ is given as a set $Cl = \{cl_1, \ldots, cl_m\}$ of clauses over atoms X. We construct a CAF $CF = (A, R, claim)$ with $A = Cl \cup X \cup \{\bar{x} \mid x \in X\}$; $R = \{(x, cl_i) \mid x \in cl_i\} \cup \{(\bar{x}, cl_i) \mid \neg x \in cl_i\} \cup \{(x, \bar{x}) \mid x \in X\}$; and $claim(cl_i) = \bar{\varphi}$ for $cl_i \in Cl$, $claim(x) = x$ for $x \in X$ and $claim(\bar{x}) = \bar{x}$ for $x \in X$. See Figure 3 for an example. It holds that φ is satisfiable iff the claim $\bar{\varphi}$ is not skeptically accepted in CF. This yields a reduction from UNSAT to $Skept_{naive}^{CAF}$ and we obtain coNP-hardness. $\square$

Hence, for CAFs in general we witness an increasing complexity for the verification problem. Interestingly, this is not the case if we restrict ourselves to well-formed CAFs. However, the higher complexity of $Skept_{naive}^{CAF}$ remains.

Theorem 2. *The complexity results for well-formed CAFs as depicted in Table 2 hold (C-c denotes completeness for complexity class C).*

Proof. Let us first consider the hardness results. The well-formed CAF problems generalise the corresponding problems for AFs and thus are as least as hard. It only remains to give a lower bound for $Skept_{naive}^{wf}$. To this end consider the CAF constructed to show the coNP-hardness of $Skept_{naive}^{CAF}$ in the proof of Proposition 3. It is easy to see that this CAF is

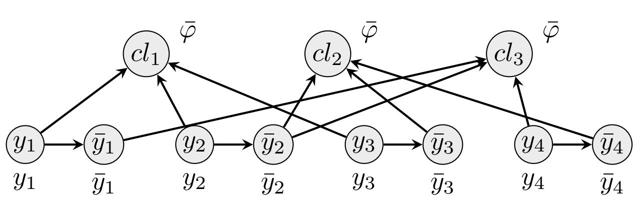


Figure 3: CAF from the proof of Prop. 3 for the formula φ with clauses $\{\{y_1, y_2, y_3\}, \{\bar{y}_2, \bar{y}_3, \bar{y}_4)\}, \{\bar{y}_1, \bar{y}_2, y_4\}\}$.

Table 2: Complexity of well-formed CAFs. Results that deviate from general CAFs (cf. Table 1) are highlighted in bold-face.

σ	$Cred_{\sigma}^{wf}$	$Skept_{\sigma}^{wf}$	Ver_{σ}^{wf}	$NEmpty_{\sigma}^{wf}$
cf	in P	trivial	in **P**	in P
naive	in P	coNP-c	in **P**	in P
grd	P-c	P-c	P-c	in P
stb	NP-c	coNP-c	in **P**	NP-c
adm	NP-c	trivial	in **P**	NP-c
com	NP-c	P-c	in **P**	NP-c
prf	NP-c	Π_2^{P}-c	**coNP**-c	NP-c

always well-formed (the only arguments that share a claim are the arguments cl_i which have no outgoing attacks) and we thus obtain that $Skept_{naive}^{wf}$ is coNP-hard.

Concerning the upper bounds we have that well-formed CAFs are a special case of CAFs and thus all the upper bounds from Theorem 1 transfer to well-formed CAFs. It only remains to give the improved upper bounds for the verification problems Ver_{σ}^{wf}: given $CF = (A, R, claim)$, to verify that $C \in \sigma_c(CF)$ we have to find a set $E \in \sigma((A, R))$ with $claim(E) = C$.

First consider admissibility based semantics: Here we first compute a maximal admissible set E of (A, R) with $claim(E) = C$. We will see then that E is unique in this sense. We start with $E_0 = \{a \in A \mid claim(a) \in C\}$. In the next step we remove from E_0 all arguments attacked by E_0 in (A, R). The resulting set E_1 is obviously conflict-free in (A, R). Now let E_2 contain all arguments from E_1 which are defended by E_1 in (A, R). We show that either $E = E_2$ or there is no admissible set E' with $claim(E') = C$. We exploit the fact that CF is well-formed. If $claim(E_2) = C$, then E_0, E_1, and E_2 attack the same arguments in (A, R) and thus $E = E_2$ is admissible in (A, R). Moreover, for each admissible set E' with $claim(E') = C$, $E' \subseteq E_2$ since all arguments attacked by E_0 in (A, R) are attacked by E' as well, and arguments not defended by E_1 in (A, R) cannot be defended by E' either. Thus, if $C \not\subseteq claim(E_2)$, there is no such E' with $claim(E') = C$ being admissible in (A, R).

If we are interested in complete semantics we additionally check whether E_2 is complete in (A, R) or not, which can be done in polynomial time. Notice that if E_2 defends an argument $a \in A$ that is not in E_2 then $claim(a) \notin C$ and moreover each set E' with $claim(E') = C$ defends a in (A, R). For preferred semantics we test in coNP whether E_2 is preferred in (A, R) or not. Notice that if there is an admissible set D with $E_2 \subset D$ then D must contain an argument a with $claim(a) \notin C$. For stable semantics we test whether E_2 is a stable extension. This can be done in P.

Now consider conflict-free and naive semantics. Take E_1 as constructed above. If $claim(E_1) = C$ we have found our conflict-free set E with $claim(E) = C$. Otherwise there is no conflict-free set E with $claim(E) = C$. To decide whether $C \in naive_c(CF)$ one additionally tests whether E is a naive extension of (A, R) (known to be in P). $\square$

Analysing the Tractability Frontier

Most of the problems considered in the previous section are computationally intractable while the importance of efficient algorithms is evident. For AFs there is a line of research to overcome the complexity of hard problems by considering special graph classes or certain parameters that characterise the structure of the AF. In what follows, we consider those problems which we have identified to be computationally hard and examine potential tractable fragments and graph parameters. Given the evident coincidence of $NEmpty_\sigma^{CAF}$ with the corresponding decision problem in AFs, we restrict ourselves here to $Cred_\sigma^{CAF}$, $Skept_\sigma^{CAF}$, and Ver_σ^{CAF}, and the corresponding problems for well-formed CAFs.

Exploiting Special Graph Classes

First, we consider graph classes that have been successfully used to obtain tractability results for AFs. Indeed every AF instance that allows to efficiently compute all extensions, also allows for efficient processing of CAF instances based on that AF. This applies to the graph classes of acyclic and noeven CAFs (Dunne 2007; Dvořák 2012), i.e. to CAFs built on top of an AF that has no directed cycle (no directed cycle of even length, respectively).

Proposition 4. *For $\sigma \in \{stb, adm, com, prf\}$, $Cred_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ are* P*-complete, and Ver_σ^{CAF} is in* P *for acyclic CAFs and CAFs without even cycles.*

Interestingly, the complexity of conflict-free semantics is not affected by these classes.

Proposition 5. *$Skept_{naive}^{CAF}$ remains* coNP*-hard and Ver_{naive}^{CAF}, Ver_{cf}^{CAF} remain* NP*-hard for acyclic CAFs.*

In fact, the CAF in the proof of Proposition 3 is acyclic; it is also well-formed, hence $Skept_{naive}^{CAF}$, and even $Skept_{naive}^{wf}$, remains coNP-hard for acyclic CAFs. The CAFs used in the proof of Proposition 1 can be made acyclic while maintaining all conflicts, which is sufficient for naive and conflict-free semantics. Hence, Ver_{naive}^{CAF}, Ver_{cf}^{CAF} are NP-hard even for acyclic CAFs.

Another prominent subclass are symmetric AFs (Coste-Marquis, Devred, and Marquis 2005). Tractability results are based on the observation that on symmetric and irreflexive AFs, stable and preferred semantics coincide with naive semantics, and admissible semantics coincides with conflict-free semantics. Thus, for this particular class of frameworks, $Cred_\sigma^{CAF}$ becomes tractable for all semantics.

Proposition 6. *$Cred_\sigma^{CAF}$ is in* P *for symmetric and irreflexive CAFs and $\sigma \in \{naive, stb, adm, com, prf\}$.*

However, in contrast to AFs, with CAFs we have that $Skept_{naive}^{CAF}$ is coNP-hard and thus the tractability argument for preferred and stable semantics breaks.

Proposition 7. *For $\sigma \in \{naive, stb, prf\}$, $Skept_\sigma^{CAF}$ is* coNP*-complete and Ver_σ^{CAF} is* NP*-complete, even for symmetric and irreflexive CAFs.*

The above is by a variant of the reduction in the proof of Proposition 3 where all attacks are made symmetric and the fact that the CAF constructed in the proof of Proposition 1 is symmetric. Note that these CAFs are not well-formed. Indeed, for well-formed CAFs we see a drop of the complexity.

Proposition 8. *For $\sigma \in \{naive, stb, prf\}$, $Skept_\sigma^{wf}$ and Ver_σ^{wf} are in* P *for symmetric and irreflexive CAFs.*

Proof. Recall that the three semantics coincide on the class of symmetric and irreflexive AFs. As Ver_{naive}^{wf} is in P (cf. Theorem 2) we obtain that Ver_{prf}^{wf} is in P as well. Likewise, for $Skept_\sigma^{wf}$ it suffices to consider naive semantics. As the CAF is well-formed, we have that arguments with the same claim are isomorphic, i.e. attack the same arguments and are attacked by the same arguments. Thus, for a naive extension we have that it either contains all arguments with a specific claim or none. To check whether a claim c is skeptically accepted we simply check whether there is an argument a that attacks the arguments with claim c. If so then c is not skeptically accepted as there is a naive extension containing a and thus not containing any argument with claim c, otherwise c is obviously skeptically accepted as their corresponding arguments are not attacked. □

The final class we consider here are bipartite AFs, for which efficient computation of the credulously and the skeptically accepted arguments is possible (Dunne 2007). This can be directly exploited for $Cred_\sigma^{CAF}$.

Proposition 9. *$Cred_\sigma^{CAF}$ is* P*-complete for bipartite CAFs and $\sigma \in \{stb, adm, com, prf\}$.*

However, as discussed earlier, skeptically accepted arguments cannot be directly applied to decide $Skept_\sigma^{CAF}$.

Proposition 10. *For $\sigma \in \{naive, stb, prf\}$, $Skept_\sigma^{CAF}$ is* coNP*-complete even for bipartite well-formed CAFs.*

The coNP-hardness can been shown by applying the reduction from the proof of Proposition 3 to the monotone 3-SAT problem which yields bipartite CAFs.

Proposition 11. *For $\sigma \in \{cf, naive, stb, adm, com, prf\}$, Ver_σ^{CAF} is* NP*-complete for bipartite CAFs.*

Proof. The AF used in the proof of Proposition 1 showing that Ver_σ^{CAF} is NP-hard is bipartite. For the NP membership of preferred semantics recall that bipartite AFs are coherent, i.e. their stable and preferred extensions coincide. □

In fact, the coherence property for bipartite AFs immediately shows that $Ver_{prf}^{wf} \in$ P, since $Ver_{stb}^{wf} \in$ P by Theorem 2.

Exploiting the Number of Claims

Next we investigate whether the number of different claims that appear in a CAF affects the complexity. For arbitrary CAFs, it is not very difficult to show that $Cred_\sigma^{CAF}$, $Skept_\sigma^{CAF}$ and Ver_σ^{CAF} remain as hard as in the general case even when restricting to CAFs $(A, R, claim)$ with $|claim(A)| = 2$, except for Ver_{cf}^{CAF}.

Proposition 12. *$Cred_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ maintain their full complexity for CAFs with only two claims.*

Proof. Given an instance of $Cred_\sigma^{CAF}$ or $Skept_\sigma^{CAF}$, i.e. a CAF $CF = (A, R, claim)$ and a claim c. We construct an instance $CF' = (A, R, claim')$ with $claim'(a) = c$ if $claim(a) = c$ and $claim'(a) = d$ otherwise. Then the claim c is credulously (resp. skeptically) accepted in CF iff c is credulously (resp. skeptically) accepted in CF'. □

For the verification problems the hardness can be shown via reductions from the $NEmpty_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ problems, except for cf for which verification becomes tractable.

Proposition 13. *For CAFs with two claims, Ver_σ^{CAF} maintains its full complexity for $\sigma \in \{naive, stb, adm, com, prf\}$, and is in* P *for $\sigma = cf$.*

However, if we consider well-formed CAFs the number of different claims has a crucial impact on the complexity and allows to employ concepts from parameterized complexity theory (Downey and Fellows 1999). A key observation of this approach is that many hard problems become tractable if some problem parameter is bounded by a fixed constant. If the order of the polynomial bound is independent of the parameter one speaks of *fixed-parameter tractability* (FPT). Here, we can give an FPT algorithm that scales exponential with the number k of different claims in the given CAF but only polynomial in its size n.

Theorem 3. *$Cred_\sigma^{wf}$, $Skept_\sigma^{wf}$, and Ver_{prf}^{wf} can be solved in time $O(2^k \cdot poly(n))$ for $\sigma \in \{naive, stb, adm, com, prf\}$.*

Proof. Let $CF = (A, R, claim)$. For $\sigma \in \{naive, stb, adm, com\}$, the algorithm builds on the observation that we can verify a set $C \subseteq claim(A)$ to be in $\sigma_c(CF)$ in P . The algorithm simply tests all subsets of $claim(A)$ for being a valid claim set w.r.t. σ and then tests whether the claim of interest is contained in one/none of these valid claim sets. For $\sigma = prf$, recall that the procedure for verification in the proof of Theorem 2 computes, for each $C \in adm_c(CF)$, the unique maximal admissible set E with $claim(E) = C$. That is, we first compute $adm_c(CF)$ and the corresponding admissible sets and then extract the maximal sets among the computed admissible sets and the corresponding sets C in $O(|adm_c(CF)| \cdot poly(n))$. □

Exploiting Tree-Width

Another approach from parameterized complexity theory for graph-based problems is the parameter of *tree-width* (Robertson and Seymour 1986) which intuitively measures how tree-like a graph is. Tree-width has been considered as parameter for AFs (Dunne 2007; Dvořák, Pichler, and Woltran 2012; Dvořák, Szeider, and Woltran 2012) and all main reasoning problems have been shown to be fixed-parameter tractable w.r.t. the tree-width of the AF.

We first briefly review the notion of the tree-width of a graph and then discuss its applications to CAFs.

Definition 5. *Let $G = (V, E)$ be a graph. A* tree decomposition *of G is a pair $(\mathcal{T}, \mathcal{X})$ where $\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T})$ is a tree and $\mathcal{X} = (X_t)_{t \in V_\mathcal{T}}$ is a set of so-called bags, which has to satisfy the following conditions:*

1. *$\bigcup_{t \in V_\mathcal{T}} X_t = V$, and for each $(v_i, v_j) \in E$, $\{v_i, v_j\} \subseteq X_t$ for some $t \in V_\mathcal{T}$.*
2. *for each $v \in V$, the subgraph of $\mathcal{T}$ induced by $\{t \mid v \in X_t\}$ is connected.*

The width of $(\mathcal{T}, \mathcal{X})$ is given by $\max\{|X_t| \mid t \in V_\mathcal{T}\} - 1$. The tree-width *of a graph G, $\mathrm{tw}(G)$, is the minimum width over all tree decompositions of G.*

As AFs can be interpreted as graphs we can also consider the tree-width of an AF. For CAFs we consider the tree-width of the AF part of the CAF.

Definition 6. *The tree-width of a CAF $CF = (A, R, claim)$ is given by the tree-width of the AF (A, R). We also write $\mathrm{tw}(CF)$ to denote the tree-width of CF.*

Courcelle's Theorem provides a powerful tool to analyse NP-hard problems. It states that any property over graphs which can be expressed in Monadic Second-Order Logic (MSO), can be decided in linear time for graphs which have bounded tree-width.

Theorem 4. *[(Bodlaender 1996; Courcelle 1987; 1990)] For every fixed MSO formula φ and integer c, there is a linear-time algorithm that, given a graph (V, E) of tree-width $\leq c$, decides whether $(V, E) \models \varphi$.*

We next show that certain results for AFs extend to CAFs.

Theorem 5. *$Cred_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ are fixed-parameter tractable w.r.t. the tree-width of the CAF for $\sigma \in \{naive, stb, adm, com, prf\}$.*

Proof. We reuse MSO characterisations for the AF extensions from (Dvořák, Szeider, and Woltran 2012). Consider CAF $CF = (A, R, claim)$ and let $\sigma_{MSO}(E)$ be an MSO formula characterising all σ-extensions of the AF (A, R). Moreover, let $I_c(.)$ be the unary predicate containing all arguments with claim c. Then we can give MSO formulas for credulous and skeptical reasoning as follows.

$$Cred_\sigma^{CAF} : \exists E \subseteq A : \sigma_{MSO}(E) \wedge \exists a \in E : I_c(a)$$

$$Skept_\sigma^{CAF} : \forall E \subseteq A : \sigma_{MSO}(E) \rightarrow \exists a \in E : I_c(a)$$

By Theorem 4, $Cred_\sigma^{CAF}$ and $Skept_\sigma^{CAF}$ are fixed-parameter tractable w.r.t. the tree-width of the CAF. □

As the graph representation of a CAF does not encode the $claim$ function explicitly, a fixed MSO characterisation for verification seems hopeless. In fact, the verification problem for CAFs stays NP-hard for constant tree-width graphs.

Proposition 14. *Ver_σ^{CAF} is* NP*-hard for graphs of tree-width 1 and $\sigma \in \{naive, stb, adm, com, prf\}$.*

Proof. Reconsider the hardness construction in the proof of Proposition 1 showing that Ver_σ^{CAF} is NP-hard. It is known that 3-SAT is NP-hard even for formulas where each variable occurs at most 3 times. If we apply the construction to such an instance we have that each connected component of the graph is of size at most 3 and bipartite. Thus the tree-width of the constructed instances is 1. □

For well-formed CAFs, the situation is again different. One can show that Ver_{prf}^{wf} (which is NP-hard for well-formed CAFs in general) is fixed-parameter tractable w.r.t. tree-width by exploiting that Ver_{prf}^{AF} is fixed-parameter tractable w.r.t. the tree-width (Dvořák, Szeider, and Woltran 2012).

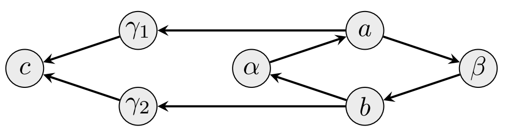
Proposition 15. *Ver^{wf}_{prf} is fixed-parameter tractable w.r.t. the tree-width of the CAF.*

Exploiting a New Parameter for Well-Formed CAFs

Our final results are based on a novel parameterisation which takes the structure of the CAF together with the distribution of the claims into account. The idea is formally captured by an *incidence graph* of a well-formed CAF, which contains both arguments and claims as vertices.

Definition 7. *The directed* incidence graph *of a CAF $CF = (A, R, claim)$ is defined as $G_{CF} = (V, E)$ with $V = A \cup claim(A)$ and $E = \{(a, claim(a)) \mid a \in A\} \cup \{(c, a) \mid (b, a) \in R, claim(b) = c\}$. We will refer to the tree-width of the incidence graph of a CAF as the incidence tree-width.*

Example 2. *Consider the CAF from Example 1. Its incidence graph is given as follows.*



Incidence tree-width also enables FPT-results.

Theorem 6. *$Cred^{wf}_{\sigma}$, $Skept^{wf}_{\sigma}$, and Ver^{wf}_{σ} are fixed-parameter tractable w.r.t. incidence tree-width for $\sigma \in \{naive, stb, adm, com, prf\}$.*

Proof. We prove the claim via Theorem 4 and also use MSO encodings $\sigma_{MSO}(\cdot)$ as in the proof of Theorem 5. As our incidence graph does not provide the attack relation directly, we replace each reference to an edge (x, y) in $\sigma_{MSO}(\cdot)$ by $\exists c \in V : (x, c) \in E \wedge (c, y) \in E$. The encodings for $Cred^{wf}_{\sigma}$ and $Skept^{wf}_{\sigma}$ are then as in the proof of Theorem 5. Verifying whether a set S is a σ_c-extension is done by the formula $\exists E \subseteq A : \sigma_{MSO}(E) \wedge (\forall c \in S, \exists a \in E : (a, c) \in E) \wedge (\forall a \in E, \exists c \in S : (a, c) \in E)$. $\square$

The result is of interest since the tractable fragments defined by the different tree-width parameters are incomparable. In fact, our final examples are used to show that the incidence tree-width of CAFs $(A, R, claim)$ can be arbitrarily smaller than the tree-width of the AF (A, R) and vice versa.

Example 3. *Consider bipartite well-formed CAFs $CF_k = (A, R, claim)$ with $A = \{b'\} \cup \{a_i, d_i \mid 1 \leq i \leq k\}$, $R = \{(a_i, b'), (a_i, d_j), (b', a_i) \mid 1 \leq i, j \leq k\}$, and with $claim(a_i) = a$, $claim(b') = b$ and $claim(d_i) = d$. The tree-width of CF_k increases with k, i.e. $\mathrm{tw}((A, R)) \geq k-1$, since we have a k-clique as graph minor. But as we only use 3 claims and deleting the claims leaves only isolated vertices in G_{CF_k}, the incidence tree-width of CF_k is ≤ 3.* $\diamondsuit$

Example 4. *Consider the well-formed CAFs $CF_k = (A, R, claim)$ with $A = \{x_i, y_{i,j} \mid 1 \leq i, j \leq k, i \neq j\}$ and $R = \{(x_i, y_{i,j}) \mid 1 \leq i, j \leq k, i \neq j\}$. As there are no undirected cycles in (A, R), the tree-width of CF_k is 1. Let $claim(x_i) = c_i$ and $claim(y_{i,j}) = claim(y_{j,i}) = c_{\max(i,j),\min(i,j)}$. Now one can show that the incidence tree-width of CF_k depends on k, i.e. $\mathrm{tw}(G_{CF_k}) \geq k - 1$, as G_{CF_k} has a k-clique as graph minor.* $\diamondsuit$

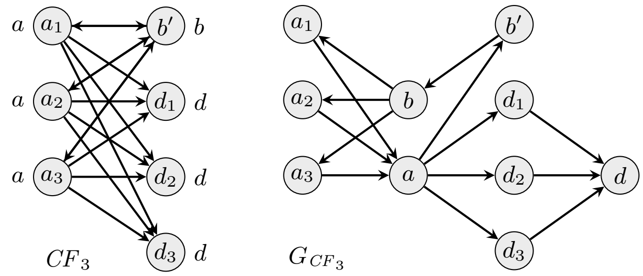


Figure 4: CAF CF_3 from Example 3 and its incidence graph.

Proposition 16. *We have (a) for each $c > 0$ there is a CAF CF with $\mathrm{tw}(CF) \geq c \cdot \mathrm{tw}(G_{CF})$, and (b) for each $c > 0$ there is a CAF CF with $\mathrm{tw}(G_{CF}) \geq c \cdot \mathrm{tw}(CF)$.*

Discussion

Related Work. Baroni, Governatori, and Riveret (2016) propose multi stage labelling systems on top of argumentation systems. They model different ways from argument acceptance to statement justification (see also (Baroni et al. 2016)), and distinguish between argument- and statement-focused approaches to argumentation; the latter amounts to our claim-based view. Their multi-labelling systems encompass several statement justification strategies from the literature and allow for a systematic comparison of structured formalisms such as ASPIC$^+$ (Modgil and Prakken 2014) and ABA (Toni 2014). With a different purpose, Corsi and Fermüller (2017) introduce semi-abstract AFs in order to build a logic of argumentation which is based on the formula of the claims. Work on rationality postulates (Caminada and Amgoud 2007; Amgoud and Besnard 2013) also takes the claim-based view but studies notions like consistency and closure of claims jointly appearing in an extension.

Generally speaking, while the interplay between arguments and their claims has been studied in the literature, there has been no systematic analysis of the computational complexity when shifting from an argument-focused to a claim-focused view, as done in this paper. We note that set variants of acceptance problems (see, e.g., (Dunne 2007)) are different as they ask whether a set of arguments is contained in an extension or in all extensions. In contrast, the claim-based setting asks whether at least one of the arguments with specific claim c is in an extension, or whether each extension contains an argument with claim c. In other words, the acceptance problems studied in (Dunne 2007) are of *conjunctive* (w.r.t. a given set of arguments) nature, while the acceptance problems we studied here are *disjunctive* w.r.t. the set of arguments with the same claim.

Summary and Outlook. In this work, we have given a complexity analysis of decision problems in abstract argumentation that are concerned with claims of arguments rather than arguments themselves. We have shown that some problems become harder under this particular view, but this effect is nearly completely mitigated when the structure of the framework is well-formed, i.e. follows some fundamental principles which are common in instantiation-based argumentation. Clarifying the complexity of the problems studied in this paper is indispensable in order to understand to which extent abstract argumentation engines can be applied within the instantiation model of argumentation. Directions for future work are: (1) studying other – less restricted – concepts than being well-formed which are tailored to particular instantiation models; (2) extending our results to justification statuses of claims that take into account a contrary relation between claims (Baroni, Governatori, and Riveret 2016); (3) studying translations that allow to (efficiently) solve claim-centric reasoning tasks with standard (i.e. argument-centric) systems; and (4) investigating the relations between parameterized complexity results for logic programs and CAFs, in particular whether there are results for LPs that can be lifted to the CAF setting.

Acknowledgements. This work is supported by the Austrian Science Fund (FWF): grants I2854, P30168 and Y698.

References

Amgoud, L., and Besnard, P. 2013. Logical limits of abstract argumentation frameworks. *Journal of Applied Non-Classical Logics* 23(3):229–267.

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

Baroni, P.; Governatori, G.; Lam, H.; and Riveret, R. 2016. On the justification of statements in argumentation-based reasoning. In Proc. *KR 2016*, 521–524. AAAI Press.

Baroni, P.; Caminada, M.; and Giacomin, M. 2011. An introduction to argumentation semantics. *Knowledge Eng. Review* 26(4):365–410.

Baroni, P.; Governatori, G.; and Riveret, R. 2016. On labelling statements in multi-labelling argumentation. In Proc. *ECAI 2016*, FAIA 285, 489–497. IOS Press.

Bodlaender, H. L. 1996. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.* 25(6):1305–1317.

Caminada, M., and Amgoud, L. 2007. On the evaluation of argumentation formalisms. *Artif. Intell.* 171(5–6):286–310.

Caminada, M.; Sá, S.; Alcântara, J.; and Dvořák, W. 2015. On the equivalence between logic programming semantics and argumentation semantics. *Int. J. Approx. Reasoning* 58:87–111.

Corsi, E. A., and Fermüller, C. G. 2017. Logical argumentation principles, sequents, and nondeterministic matrices. In Proc. *LORI 2017*, LNCS 10455, 422–437. Springer.

Coste-Marquis, S.; Devred, C.; and Marquis, P. 2005. Symmetric argumentation frameworks. In Proc. *ECSQARU 2005*, LNCS 3571, 317–328. Springer.

Courcelle, B. 1987. Recognizability and second-order definability for sets of finite graphs. Technical Report I-8634, Université de Bordeaux.

Courcelle, B. 1990. Graph rewriting: an algebraic and logic approach. In *Handbook of theoretical computer science, Vol. B*. Amsterdam: Elsevier. 193–242.

Downey, R. G., and Fellows, M. R. 1999. *Parameterized Complexity*. Springer.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* 77(2):321–358.

Dunne, P. E. 2007. Computational properties of argument systems satisfying graph-theoretic constraints. *Artif. Intell.* 171(10-15):701–729.

Dvořák, W., and Dunne, P. E. 2018. Computational problems in formal argumentation and their complexity. In *Handbook of Formal Argumentation*. College Publications. 631–687.

Dvořák, W.; Pichler, R.; and Woltran, S. 2012. Towards fixed-parameter tractable algorithms for abstract argumentation. *Artif. Intell.* 186:1 – 37.

Dvořák, W.; Szeider, S.; and Woltran, S. 2012. Abstract argumentation via monadic second order logic. In Proc. *SUM 2012*, LNCS 7520, 85–98. Springer.

Dvořák, W. 2012. *Computational Aspects of Abstract Argumentation*. Ph.D. Dissertation, Vienna University of Technology, Institute of Information Systems.

Gorogiannis, N., and Hunter, A. 2011. Instantiating abstract argumentation with classical logic arguments: Postulates and properties. *Artif. Intell.* 175(9-10):1479–1497.

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

Modgil, S., and Prakken, H. 2018. Abstract rule-based argumentation. In *Handbook of Formal Argumentation*. College Publications. 287–364.

Prakken, H., and Vreeswijk, G. A. 2002. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*, volume 4. Springer.

Robertson, N., and Seymour, P. D. 1986. Graph minors. II. Algorithmic aspects of tree-width. *J. Algorithms* 7(3):309–322.

Toni, F. 2014. A tutorial on assumption-based argumentation. *Argument & Computation* 5(1):89–117.

Wu, Y.; Caminada, M.; and Gabbay, D. M. 2009. Complete extensions in argumentation coincide with 3-valued stable models in logic programming. *Studia Logica* 93(2-3):383–403.