TU WIEN Informatics

Graph-Classes of Argumentation Frameworks with Collective Attacks

Properties and Complexity Results

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Logic and Computation

eingereicht von

Matthias König, BSc
Matrikelnummer 01425517

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Dr.techn. Wolfgang Dvořák
Mitwirkung: Univ.Prof. Stefan Woltran

Wien, 24. August 2020

Matthias König

Wolfgang Dvořák

Graph-Classes of Argumentation Frameworks with Collective Attacks

Properties and Complexity Results

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Logic and Computation

by

Matthias König, BSc
Registration Number 01425517

to the Faculty of Informatics

at the TU Wien

Advisor: Dr.techn. Wolfgang Dvořák
Assistance: Univ.Prof. Stefan Woltran

Vienna, 24th August, 2020

Matthias König — Wolfgang Dvořák

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Matthias König, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 24. August 2020

Matthias König

Acknowledgements

Throughout the work of this thesis and my studies as a whole I have received a lot of support and assistance.

First of all, I would like to thank my advisors Wolfgang Dvořák and Stefan Woltran for their support and encouragement, and for introducing me to a friendly and welcoming working environment. Their frequent and detailed feedback and insightful remarks made this thesis possible.

I would also like to thank all colleagues who accompanied me during my studies. Special thanks go to Michael Bernreiter, who gave me invaluable feedback and who I can always rely on. Moreover I want to thank (in random order) Timothée Durand, David Penz, Giulio Pace, Tamara Drucks, Marcel Moosbrugger, Hanna Lachnitt, Grace Longo, Pamina Georgiou, Thomas Hader, and all the members of the BT-group. Their company and cooperation made the last years as good as they were.

Finally, I want to thank my family and friends for their support and patience, and for enabling me to pursue my studies the way I wanted.

Kurzfassung

Abstract Argumentation Frameworks (AFs) bieten die Möglichkeit, Wissen auf einfache Art zu repräsentieren und Schlüsse daraus zu ziehen. Diese Einfachheit führte dazu, dass sie seit ihrer Einführung durch Phan Minh Dung im Jahr 1995 immer beliebter wurden, und in zahlreichen wissenschaftlichen Publikationen zur Anwendung in verschiedenen Bereichen verwendet wurden. Die simple Struktur der Frameworks - das Wissen wird in Form von Argumenten und Attacken als gerichteter Graph dargestellt - lässt allerdings bestimmte syntaktische Konstruktionen nicht zu, weshalb in der Folge einige Erweiterungen für den Formalismus eingeführt wurden. Diese Arbeit beschäftigt sich mit so einer Erweiterung: *Argumentation Frameworks mit gemeinsamen Attacken* (SETAFs). Hier können Attacken - anstatt nur von einem Argument - im Allgemeinen von beliebig vielen Argumenten ausgehen. Die zugrundeliegende Struktur ist damit kein gerichteter Graph mehr, sie ist ein *gerichteter Hypergraph*.

Die Vielzahl an Anwendungsmöglichkeiten bedingt die Suche nach effizienten Argumentationssystemen. Die Komplexität der meisten gängigen Probleme auf Argumentation Frameworks ist auf den ersten beiden Stufen der polynomiellen Hierarchie einzuordnen, weshalb eine detaillierte Komplexitätsanalyse erforderlich ist. In dieser Arbeit wird nach Fragmenten mit geringerer Komplexität gesucht. Dazu werden Strukturen in den Hypergraphen, welche den SETAFs zugrunde liegen, betrachtet. Der Fokus liegt herbei auf Azyklizität, Symmetrie, und 2-Färbbarkeit bzw. Bipartitness – es werden verschiedene Varianten dieser Eigenschaften für Hypergraphen definiert und geprüft, ob die entsprechenden Klassen von SETAFs eine Reduktion der Komplexität bei den gängigen Argumentationsproblemen aufweisen. Es werden die folgenden einfachen Fragmente etabliert: SETAFs, welche keine Zyklen beinhalten, die aus mehr als einem Argument bestehen, SETAFs ohne Zyklen gerader Länge, ϵ-symmetrische SETAFs ohne Selbst-Attacken, und β-bipartite SETAFs. Die entsprechenden Eigenschaften werden in dieser Arbeit eingeführt.

Abstract

Abstract Argumentation Frameworks (AFs) constitute a simple formalism to represent and reason over knowledge. The simple way of modelling information just as a directed graph lead to a rapid rise of its popularity. However, the simplicity came with the cost of strict syntactic restrictions, and as it is often inconvenient to formalize certain natural structures in the original notion of frameworks, many extensions have been proposed. We deal with one such extension, namely Argumentation Frameworks with *Collective Attacks* (SETAFs), which allow attacks to origin from sets of arguments. In particular, this means that instead of a directed graph, the underlying structure of the framework is a *directed hypergraph*.

Many applications motivated the search for efficient reasoning tools for the frameworks, and as most reasoning problems are intractable in general, a more fine-grained complexity analysis was needed. In this thesis we try to identify *tractable fragments* for reasoning tasks on SETAFs. In particular, we investigate how certain restrictions on the hypergraph-structure such as acyclicity, symmetry, and bipartiteness can be formulated and whether they allow us to reason more efficiently. We show that the low complexity of acyclicity carries over from AFs to SETAFs, and give the even more general result that this fragment extends to frameworks that have cycles of length 1. Moreover we establish tractable fragments in even-cycle-freeness, self-attack-free ϵ-symmetry, and β-bipartiteness, the respective defining properties are introduced in the course of this thesis.

Contents

CHAPTER 1

Introduction

> "The ability to engage in arguments is essential for humans to understand new problems, to perform scientific reasoning, to express, clarify and defend their opinions in their daily lives." [Dun95]

1.1 Argumentation in AI

Argumentation is needed to resolve conflicts in potentially inconsistent or incomplete knowledge, which is essential to draw conclusions of any kind. In both single-agent systems and distributed systems consisting of multiple entities acting on their own, argumentation techniques are used to evaluate information and draw coherent conclusions. While in the former the agent will use them to decide the next steps, in the latter conflicts between different agents can be resolved to reach a consensus. For more insights on the importance of Argumentation on Artificial Intelligence the reader is referred to [BD07].

Abstract argumentation offers a uniform framework to solve various tasks of this kind, independent of the actual nature of the area of application. Abstract Argumentation Frameworks (AFs), introduced in his influential seminal paper [Dun95], are Dung's approach to provide a versatile system for reasoning tasks in an intuitive setting.

In AFs we view arguments just as nodes in a directed graph, independent from their internal structure. The directed edges are referred to as "attacks" and model conflicts between the arguments. In this thesis we will mainly consider Argumentation Frameworks with collective attacks (SETAFs), a generalization of AFs introduced by Nielsen and Parsons in [NP06]. SETAFs allow a more expressive syntax, in particular the natural notion of attacks that require more than one argument can be directly expressed with SETAFs, whereas in the original frameworks this requires unintuitive additional constructions.

Figure 1.1: (a) An AF and (b) a SETAF with a collective attack $(\{a, b\}, c)$. Note that the attack on c in (a) is effective if at least one of a and b is accepted, whereas the attack on c in (b) requires the acceptance of *both* a and b to be effective.

For an illustration of an AF and a SETAF see Figure 1.1, for a detailed overview of these formalisms the reader is referred to [Dun95] and [FB19] respectively.

This simple way of representing arguments allows also "non-experts" to formalize problems from virtually all domains. Applications for Argumentation Frameworks range from more theory-driven results such as an alternative formulation of default logic [BDKT97] or studies of dialogues [JV99] to more practical issues such as argumentation based machine learning [MZB07], defeasible reasoning and conflicting norms [Pra11], and legal reasoning in general [BPS09].

As a non-monotonic formalism it is a suitable tool to represent the intuitive argumentation mode of human interaction. Moreover, it is conveniently close to other well-studied non-monotonic formalisms such as logic programming, defeasible logic, and inductive defeasible logic, which was already pointed out in [Dun95]. This allows us to use existing solvers for these formalisms to evaluate argumentation frameworks.

The abstract reasoning process is illustrated in Figure 1.2. We extract arguments and attacks (i.e. potential conflicts between the arguments) from an existing knowledge base (KB) and build an abstract representation of the knowledge, then we apply the reasoning procedure on the abstract level. Finally, we translate the results back to the actual domain and draw conclusions.

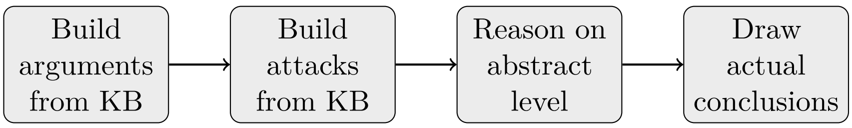


Figure 1.2: Illustration of argumentation process.

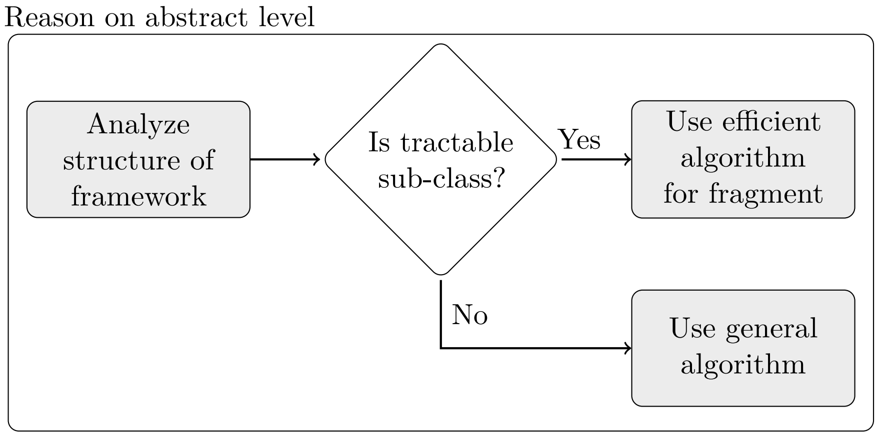


Figure 1.3: Illustration of the refined abstract reasoning process

The reasoning process (the third step in Figure 1.2), in general, is intractable, which encourages more detailed complexity results. In particular, the standard reasoning problems that we consider in this thesis - credulous and skeptical reasoning - are often not efficiently computable. One approach to overcome this is to identify *tractable fragments*, i.e. classes of problems that allow us to use more efficient algorithms [DB01, Dun07, CMDM05, DD17]. We can exploit them in two ways: either it is already known that the knowledge base yields certain structural properties because of the way it is obtained, in this case we can apply the respective more efficient algorithms right away.

If we cannot a priori guarantee such properties, it might provide a computational speed-up to preprocess the abstract framework and (efficiently) check for the requirements for the applicability of the efficient algorithms, and only perform them if they are met. This refined procedure is illustrated in Figure 1.3. In this thesis we will identify such tractable fragments for SETAFs.

SETAFs are evaluated w.r.t. *semantics* which determine which abstract arguments are coherently acceptable. Since Argumentation Frameworks have been introduced, in the last years various different semantics have been suggested and studied. We will examine conflict-free, admissible, grounded, naive, stable, complete, preferred, stage, and semi-stable semantics for SETAFs (cf. [NP06, DGW18, FB19]).

1.2 Main Contributions

Tractable fragments have been studied for AFs [DB01, Dun07, CMDM05, DD17]. However, the more general notion of SETAFs has not yet been analysed in detail when it comes to classes that yield a lower complexity. We will define natural generalizations of already discovered graph-classes that form tractable fragments, as well as more involved notions that are motivated by results from the hypergraph literature. In particular, we will study three different families of graph-properties thoroughly and suggest suitable definitions:

- We will examine the properties of SETAFs that only have cycles up to or at least of a certain length, only cycles of a certain parity (even or odd), or no cycles at all. Moreover, we will adapt notions of acyclicity that origin in the hypergraph literature for SETAFs, in particular definitions used in the field of database theory.
- Regarding symmetry we will not only provide natural notions of symmetric SETAFs, but also discuss properties of symmetric directed graphs that do not hold for hypergraphs and give negative results, indicating which generalizations are not suitable for symmetric SETAFs. As for directed hypergraphs symmetry has not been a field of extensive studies, these will be novel definitions that are motivated by characteristics of SETAFs rather than hypergraphs in general.
- Adapting existing results from AFs, we will provide a way to exploit 2-colorability in SETAFs, after giving different natural definitions of bipartiteness for SETAFs. Moreover we will show how in bipartite SETAFs a partitioning can be computed efficiently, allowing us to apply our discoveries in a practical context by providing an algorithm.

Beside general properties we will investigate the complexity of reasoning problems for these classes and identify tractable fragments. In particular, when we identify a class of SETAFs that allows us to reason efficiently, we will show that it is also possible to efficiently decide the respective property, which is essential to exploit the property.

1.3 Structure of the Thesis

This thesis is organized in 6 chapters. The first is this introduction, where we explain the importance of Argumentation in Artificial Intelligence, describe the main contributions and structure of this thesis. In Chapter 2 we introduce the notions we will use in the later chapters, in particular we give definitions for Argumentation Frameworks and Complexity Theory, and we explain how they relate to each other.

In Chapter 3 we discuss different cyclic or acyclic SETAFs and how different cycle-related properties such as acyclicity, minimal and maximal cycle length, and the absence of cycles of a certain parity affect the complexity of the reasoning problems defined in the preliminaries.

Then, in Chapter 4 we discuss whether symmetry allows us to reason more efficiently on SETAFs, as it does on AFs. We will introduce three suitable notions of symmetry with different properties and examine the complexity of credulous and skeptical reasoning in the respective classes of SETAFs.

After that, in Chapter 5 we will adapt bipartiteness and 2-colorability for SETAFs and state properties and complexity results for the resulting classes of SETAFs. Finally, in Chapter 6 we will summarize the results from the previous chapters and give an outlook to future work, state problems that are still open, and mention related work.

CHAPTER 2

Preliminaries

In this chapter we will recapitulate the basic notions and definitions that are needed for the main part of this thesis. We will start with Complexity Theory in Section 2.1, where a very brief introduction to the concepts of Turing machines, computational complexity classes, and reducibility is given. Moreover we will introduce some fundamental problems that will later on be used to obtain complexity results.

Then we establish the basics of Argumentation Theory in Section 2.2, in particular by means of Argumentation Frameworks. Among other things, we will see the semantics that we will examine in this thesis and fix some semantic and syntactic terms regarding Argumentation Frameworks.

Subsequently, in Section 2.3 we will connect the complexity part and the argumentation part by giving an overview of existing complexity results for reasoning problems on Argumentation Frameworks and some standard approaches to establish such.

2.1 Complexity Theory

In this section we will briefly introduce the basic notions of complexity classes and reductions, and introduce problems that are representative for their respective classes. For a comprehensive introduction to complexity theory the reader is referred to [Pap94], which also introduces most of the notation we are going to use in the following.

Complexity classes can be defined in terms of *Turing machines* which were first introduced by Turing in [Tur36]. We will not go into detail with Turing machines; intuitively they consist of an input tape and a fixed number of working tapes that each consist of cells that contain symbols from a predefined alphabet, a finite set of states, one of which is called "yes" state, the others are called "no" states, and a transition function that determines the movement of a read/write head for each tape that manipulates the symbols on the tapes depending on the state and the symbols that are on the current position of the read/write heads of each tape. If for each combination of state and tape symbols there is at most one successive step, we call the Turing machine *deterministic*, if there is at least one such transition where there are more than one possible successive steps we call it *non-deterministic*. If for a configuration there is no successive step w.r.t. the transition function, the Turing machine is said to *halt*, i.e. it is in a *halting state*.

The problems we are considering are *decision problems*, that is the answer to the question for a given input (i.e. an initial configuration on the input tape of a Turing machine) after all steps are carried out as specified by the transition function, does the Turing machine halt in the "yes" state. As all problems we deal with are decidable, we are mainly concerned with Turing machines that are guaranteed to halt, either in the "yes" state (in which case the input is said to be a "yes" instance of the respective problem) or in a "no" state (in which case the input is said to be a "no" instance of the respective problem). For non-deterministic Turing machines an input is a "yes" instance iff at least one possible sequence of transition steps on the input leads to the "yes" state, and a "no" instance otherwise.

We measure the complexity of a decision problem in terms of the time/space requirements for deciding the problem, i.e. the number of steps a Turing machine has to perform in the worst case or the number of tape cells that it needs to write on respectively. Here we are interested in the asymptotic worst-case behaviour depending on the size of the input, which is the number of cells on the input tape that are not empty in the initial configuration. In the following we will denote the size of the input with n. For an arbitrary decision problem P, if there is a Turing machine TM such that for every "yes" instance of this problem TM halts in the "yes" state after at most $f(n)$ steps (where $f(\cdot)$ is an arbitrary monotone function), the problem is said to be in (i.e. a member of) the respective time-complexity class (determined by f). Likewise if for every "yes" instance of this problem a Turing machine reaches the "yes" state after using at most $f(n)$ tape cells on the working tapes, the problem is said to be in (i.e. a member of) the respective space-complexity class (again, determined by f). Sometimes for a time/space complexity class $\mathcal{C}$ we say "P is $\mathcal{C}$-easy" if a problem P is a member of $\mathcal{C}$.

If there is a deterministic Turing machine that accepts (that is, halts in the "yes" state) every "yes" instance of a problem after using at most $f(n)$ steps, where f is asymptotically polynomial in n, i.e. there is an n_0 and a k such that for every $n \geq n_0$ we have $f(n) \leq n^k$, the problem is said to be in the complexity class P (polynomial time). Likewise, if the respective Turing machine is non-deterministic, the problem is said to be in the complexity class NP (non-deterministic polynomial time). If there is a deterministic Turing machine that accepts every "yes" instance of a problem after writing on most $f(n)$ tape cells, where f is asymptotically logarithmic in n, i.e. there is an n_0 and a k such that for every $n \geq n_0$ we have $f(n) \leq log_k(n)$, the problem is said to be in the complexity class L (logarithmic space).

For every problem P its co-problem $\bar{P}$ is defined such that all "yes" instances of P are "no" instances of $\bar{P}$ and all "no" instances of P are "yes" instances of $\bar{P}$. For a complexity class $\mathcal{C}$ we say that its co-class consists of all co-problems of $\mathcal{C}$.

A *reduction* from a problem P_a to a problem P_b is a function g that transforms every "yes" instance of P_a into a "yes" instance of P_b and every "no" instance of P_a into a "no" instance of P_b. If g is computable in polynomial time (or in logarithmic space if we are dealing with complexity classes below NP) w.r.t. the size of the input of problem P_a, we say the reduction is efficient; if for every problem of a complexity class $\mathcal{C}$ there is an efficient reduction to a problem P we say P is *hard* for $\mathcal{C}$ (or simply $\mathcal{C}$-hard). Reducibility is transitive, i.e. if we can efficiently reduce a problem P_a to a problem P_b and P_b to a problem P_c, we can efficiently reduce P_a to P_c. Hence, to establish hardness it suffices to reduce a problem that is known to be hard for the respective complexity class to the problem in question. If a problem P is both hard for and a member of a complexity class $\mathcal{C}$ we say it is $\mathcal{C}$-complete.

An *Oracle-Turing machine* augments the concept of a Turing machine by a dedicated *query tape* and a *query state*. Intuitively, these enable the machine to decide a predefined oracle-problem of a complexity class $\mathcal{C}$ in one step. If the Oracle-Turing machine is in the query state, after one step the Oracle-Turing machine will go in a dedicated "query-yes" state iff the content of the query tape is a "yes" instance of the oracle-problem and in a "query-no" state otherwise. We choose for an oracle problem one that is complete for the complexity class $\mathcal{C}$, this way with little adjustments the oracle can decide arbitrary problems that are in $\mathcal{C}$. By $\mathcal{C}_1^{\mathcal{C}_2}$ we denote the class of problems that are in $\mathcal{C}_1$ for Turing machines with an oracle for problems in $\mathcal{C}_2$. This allows us to define a hierarchy of complexity classes, the so called *polynomial hierarchy* [Pap94]:

$$\begin{aligned}
\mathsf{P} &= \Sigma_0^P = \Pi_0^P = \Delta_0^P \\
\Sigma_{i+1}^P &= \mathsf{NP}^{\Sigma_i^P} \\
\Pi_{i+1}^P &= \mathsf{coNP}^{\Pi_i^P} \\
\Delta_{i+1}^P &= \mathsf{P}^{\Sigma_i^P}
\end{aligned}$$

Note that an oracle for a complexity class $\mathcal{C}$ is also an oracle for its co-class. We have that Σ_i^P is the co-class of Π_i^P for $i \geq 0$. Moreover we have the following relationships between the classes that we deal with:

$$\mathsf{L} \subseteq \mathsf{P} \subseteq \begin{matrix}\mathsf{NP}\\ \mathsf{coNP}\end{matrix} \subseteq \Delta_2^P \subseteq \begin{matrix}\Sigma_2^P\\ \Pi_2^P\end{matrix} \subseteq \cdots \subseteq \Delta_i^P \subseteq \begin{matrix}\Sigma_i^P\\ \Pi_i^P\end{matrix} \subseteq \Delta_{i+1}^P \subseteq \begin{matrix}\Sigma_{i+1}^P\\ \Pi_{i+1}^P\end{matrix} \subseteq \ldots$$

None of these inclusions are known to be proper.

Now we will introduce some canonical problems that are known to be complete for their respective complexity classes; we will later use them to establish reductions to show the hardness of our problems in question. To this end we introduce conjunctive-normal-form formulas (in the following: *CNF-formulas*). A CNF-formula φ consists of a set of *clauses* C, each clause $c \in C$ is a set of *literals*, each literal is either a positive or negative atom, denoted by x or $\bar{x}$ respectively. An interpretation (assignment) $\mathcal{I}$ is a subset of the atoms occurring in a CNF-formula φ, $\mathcal{I}$ makes φ true (i.e. satisfies φ) iff for each clause c there either is a positive literal $x \in c$ such that $x \in \mathcal{I}$ or a negative literals $\bar{x} \in c$ such that $x \notin \mathcal{I}$.

Let X be the set of all atoms in a CNF-formula and let $Y \subseteq X$ be a subset thereof, a *partial assignment* I_Y is then a subset of Y. A $QBF^2_\forall$-formula $\Phi = \forall Y \exists Z\, \varphi$, where φ is a CNF-formula and Y, Z are subsets of the set of all atoms X in φ such that $Y \cup Z = X$ and $Y \cap Z = \emptyset$, is true iff for every partial assignment $\mathcal{I}_Y$ there is a partial assignment $\mathcal{I}_Z$ such that $\mathcal{I}_Y \cup \mathcal{I}_Z$ satisfy φ. The following completeness results are known:

- The problem *SAT* of deciding whether for a given CNF-formula φ there is a satisfying assignment $\mathcal{I}$ is NP-complete, the respective co-problem (i.e. to decide whether for φ there is *no* satisfying assignment) is coNP-complete.
- The problem $QSAT^2_\forall$ of deciding whether a given $QBF^2_\forall$-formula Φ is true is Π^P_2-complete, the respective co-problem (i.e. to decide whether Φ is not true) is Σ^P_2-complete.
- The problem *MINSAT* of deciding whether for a given CNF-formula φ an atom x is in some $\subseteq$-minimal model is Σ^P_2-complete, the respective co-problem (i.e. to decide whether x is not in some $\subseteq$-minimal model) is Π^P_2-complete [EG93].

These problems remain hard if we add certain restrictions, in particular we can restrict φ such that there are at least 2 clauses in φ, and each clause has to have at least one positive and at least one negative literal. These restrictions will help us later to simplify certain reductions.

2.2 Argumentation Frameworks

Argumentation Frameworks (AFs), along with the most common notions, have been introduced by Dung in his influential seminal paper [Dun95]. Since then, the basic formalism has been extended by different concepts, one of which are collective attacks. The class constituted by this generalization is called SETAFs, they were first introduced by Nielsen and Parsons in [NP06], where also the definitions we will encounter in this section were first formulated, adapting the original notions of Dung. As the thesis at hand is mostly concerned with SETAFs, we will start with the more general notion of SETAFs and then introduce AFs as a special case thereof. Even though our definition of AFs will then not be syntactically identical with the standard definition (where attacks are between two arguments only, not a set of arguments and an argument), it is very easy to translate one concept into the other, and this variation will allow us to formulate our results more easily. Consequently, we will start with the basic definitions.

Definition 1. *A* SETAF *is a pair* (A, R)*, where* A *is a set of* arguments *and* $R \subseteq (2^A \setminus \{\emptyset\}) \times A$ *is the* attack relation*.*

Let $SF = (A, R)$ be a SETAF, then we say an attack $(T, h) \in R$ is *towards* h, and we call T the *tail* of the attack and h the *head*. Moreover we say any set $T' \supseteq T$ *attacks* h, denoted by $T' \rightarrowtail_R h$. Similarly, we say T' attacks any set H with $h \in H$, again denoted by $T' \rightarrowtail_R H$. For an attack (T, h) towards h we say any set S such that S attacks T *defends* h against said attack. A set of arguments argument S is said to be *acceptable* w.r.t. a set of arguments T if T defends S against all attacks towards it. For a set $S \subseteq A$ we denote its *range* w.r.t. R by $S_R^{\oplus} = S \cup \{a \mid S \rightarrowtail_R a\}$.

If $h \in T$ we say (T, h) is a *self-attack*, if also $|T| = 1$ we say (T, h) is a self-loop, and the argument $t \in T$ is *self-looping*. An attack (T, h) with $|T| = 1$ is called an *AF-attack* and is sometimes denoted by (t, h) where $T = \{t\}$, i.e. the tail is denoted as a single argument rather than a set. This is just a matter of notation, as in our context the tail of every attack is a set of arguments.

Definition 2. *A SETAF that only consists of AF-attacks is an* Argumentation Framework *(AF).*

Towards the semantics of SETAFs, we will first define the characteristic function of a SETAF.

Definition 3. *The* characteristic function $\mathcal{F}_{SF} : 2^A \to 2^A$ *of a SETAF* $SF = (A, R)$ *is defined as* $\mathcal{F}_{SF}(S) = \{a \mid a \in A, a$ *is defended by* $S\}$*.*

We will now define the semantics of SETAFs. This is done in terms of *extensions*. An extension w.r.t. a semantics σ is a subset of the arguments of a SETAF that can be jointly accepted, that is, according to the restrictions proposed by σ, which arguments an agent could accept on the basis of the information given by the SETAF. A semantics σ yields a set of extensions for every SETAF SF, we denote this set of extensions by $\sigma(SF)$. We will introduce the semantics conflict-free, admissible, grounded, naive, stable, complete, preferred, stage, and semi-stable, abbreviated by *cf*, *adm*, *grd*, *naive*, *stb*, *com*, *pref*, *stg*, and *sem* respectively (cf. [NP06, DGW18, FB19]).

Definition 4. *Let* $SF = (A, R)$ *be a SETAF and let* $E \subseteq A$ *be a set of arguments. Then*

- $E \in cf(SF)$*, if for every attack* $(T, h) \in R$ *we have* $T \cup \{h\} \not\subseteq E$*,*
- $E \in adm(SF)$*, if* $E \in cf(SF)$ *and* E *defends itself in* SF*,*
- $E \in grd(SF)$*, if* E *is the least fixpoint of* $\mathcal{F}_{SF}$ *or equivalently* $E = \bigcap_{S \in com(SF)} S$*,*

- $E \in naive(SF)$, *if* $E \in cf(SF)$ *and there is no* $E' \in cf(SF)$ *with* $E' \supset E$,
- $E \in stb(SF)$, *if* $E \in cf(SF)$ *and* $E \rightarrowtail_R a$ *for all* $a \in A \setminus E$,
- $E \in com(SF)$, *if* $E \in adm(SF)$ *and* $a \in E$ *for all* $a \in A$ *defended by* E,
- $E \in pref(SF)$, *if* $E \in adm(SF)$ *and there is no* $E' \in adm(SF)$ *with* $E' \supset E$,
- $E \in stg(SF)$, *if* $E \in cf(SF)$ *and there is no* $E' \in cf(SF)$ *with* $E'^{\oplus}_R \supset E^{\oplus}_R$,
- $E \in sem(SF)$, *if* $E \in adm(SF)$ *and there is no* $E' \in adm(SF)$ *with* $E'^{\oplus}_R \supset E^{\oplus}_R$.

The following relationships between the semantics of SETAFs have been established in [NP06, DGW18, FB19] and generalize the respective results for AFs. For any SETAF SF we have:

$$stb(SF) \subseteq sem(SF) \subseteq pref(SF) \subseteq com(sf) \subseteq adm(SF) \subseteq cf(SF)$$

$$stb(SF) \subseteq stg(SF) \subseteq naive(SF) \subseteq cf(SF)$$

$$grd(SF) \subseteq com(SF)$$

Moreover, it is known that if SF has at least one stable extension, then the stable, semi-stable and stage semantics coincide. See Example 1 and its corresponding hypergraph structure in Figure 2.1 for an illustration of a SETAF, and see Table 2.1 for its semantics.

Example 1. *$SF = (A, R)$ with $A = \{a, b, c, d, e, f\}$ and $R = \{(a, b), (b, e), (c, b), (\{e, f, c\}, c), (d, d), (\{d, e\}, a), (f, b)\}$.*

Now we will have a look at some further possible syntactic and semantic properties of SETAFs. First we consider redundancies. In SETAFs certain attacks do not contribute to

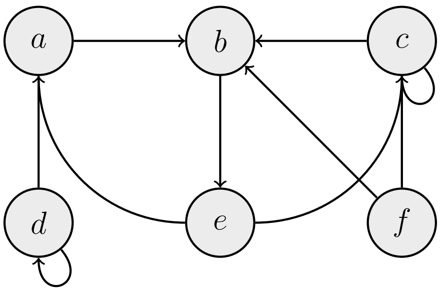


Figure 2.1: SETAF from Example 1. We will indicate collective attacks by arcs that all meet the head of the attack at the same angle (e.g. the attack $(\{d, e\}, a)$), whereas separate attacks towards the same argument meet it at different angles (e.g. the attacks (a, b) and (c, b)).

σ	$\sigma(SF)$
cf	$\{\{a,c,e\},\{c,e\},\{a,c,f\},\{a,c\},\{c,f\},\{c\},\{a,e,f\},\{e,f\},\{a,e\},\{e\},$ $\{b\},\{a,f\},\{a\},\{f\},\{\}\}$
adm	$\{\{e,f\},\{f\},\{\}\}$
grd	$\{\{e,f\}\}$
naive	$\{\{b\},\{a,e,f\},\{a,c,e\},\{a,c,f\}\}$
stb	$\{\}$
com	$\{\{e,f\}\}$
pref	$\{\{e,f\}\}$
stg	$\{\{a,e,f\},\{a,c,e\},\{a,c,f\}\}$
sem	$\{\{e,f\}\}$

Table 2.1: Semantics of the SETAF in Example 1.

the overall meaning of the framework and might therefore just as well be removed. Some of these meaningless attacks are captured by the notion of *redundancy-free* SETAFs, a class that is very easy to detect. It is computationally easy to lose these attacks, and, moreover, it yields certain advantages to restrict ourselves to redundancy-free SETAFs, as many of the proofs we will encounter in this thesis are easier to understand if we do not have to take redundant attacks into account. This notion of redundancy-freeness is from [Pol17].

Definition 5. *Let $SF = (A, R)$ be a SETAF, then an attack $(T, h) \in R$ is called* redundant *if there exists an attack $(T', h) \in R$ such that $T' \supset T$. A SETAF that contains no redundant attacks is called* redundancy-free.

By results of [Pol17] and [DRW20] we know that removing these redundant attacks does not change the semantics of the SETAF, i.e. if we have a SETAF $SF = (A, R)$ and its redundancy-free counterpart $SF' = (A, \{r \mid r \in R, r \text{ not redundant in } SF\})$ then we have $\sigma(SF) = \sigma(SF')$ for all semantics σ under our consideration.

As with the basic notions, the following definitions origin in [Dun95] and were adapted for SETAFs in [NP06].

Definition 6. *A SETAF $SF = (A, R)$ is called* well-founded *if there exists no infinite sequence of sets $B_1, B_2, \ldots,$ such that for all i, B_i is the tail of an attack towards an argument in B_{i-1}.*

Also from [NP06] we know that well-founded SETAFs have desirable properties, e.g. the complete, preferred and stable extensions coincide with the (unique) grounded extension.

Let $a1, a2, \ldots$ be a sequence of arguments, where each argument a_i is in the tail of an attack towards a_{i-1}. Then the arguments with an even index i are said to *indirectly attack* a_1, while the arguments with an odd index i are said to *indirectly defend* a_1. If an argument a both indirectly attacks and indirectly defends an argument b, then it is said to be controversial w.r.t. b.

Definition 7. *A SETAF is* limited controversial *if there exists no infinite sequence of arguments $a_1, a_2, \ldots$ such that for all i the argument a_i is controversial w.r.t. a_{i-1}.*

This leads us to the notion of coherent SETAFs. In the following we will use coherency to show that the low complexity of reasoning problems for the grounded extension in some cases carries over to other semantics.

Definition 8. *A SETAF is coherent, if its preferred and stable extensions coincide.*

Note that it is always the case that every stable extension is preferred, the class of SETAFs where also the converse holds is captured by the notion of coherency. In [NP06] it was shown that both well-foundedness and limited controversialness are sufficient conditions for coherency. As in coherent SETAFs stable and preferred extensions coincide and there always is at least one preferred extension, we also have that in coherent SETAFs the stable extensions coincide with the stage and semi-stable extensions.

We have that the structure of a SETAF is a *directed hypergraph*, where the arguments are the *vertices* and the attacks are *directed hyperedges*. As we have that the head of any attack always contains exactly 1 argument, the hyperedges are called *backward hyperarcs* or B-arcs for short, and a SETAF is a B-hypergraph, as it only consists of B-arcs. Note that in SETAFs for each edge $(T, h) \in R$ we also have $T \neq \emptyset$, which is not necessarily the case for general B-hypergraphs. In the course of this thesis we will define restrictions on the hypergraph structure of SETAFs and examine whether this allows us to reason more efficiently, i.e. yields a lower complexity. To this end we will now define some basic graph-properties, the specific graph-classes will be defined in the respective chapters.

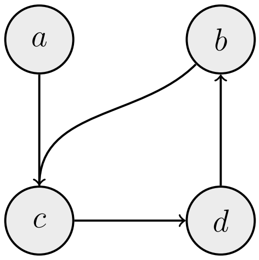
Definition 9. *Let $SF = (A, R)$ be a SETAF. A* path *P of length $|P| = n$ is a sequence of arguments and attacks*

$$P = (a_1, (S_1, a_2), a_2, (S_2, a_3), \ldots, (S_n, a_{n+1}), a_{n+1}),$$

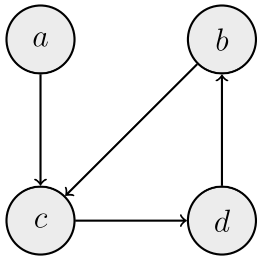
where $a_i \in S_i$ for $1 \leq i \leq n$. A path P of length n is a cycle *if $a_1 = a_{n+1}$.*

Sometimes we do not necessarily care about the attacks that a path or cycle consists of, in this case we may denote a path (cycle) $P = (a_1, (S_1, a_2), a_2, (S_2, a_3), \ldots, (S_n, a_{n+1}), a_{n+1})$ solely by the arguments that appear in the heads of the respective attacks, i.e. $P = (a_1, a_2, \ldots, a_{n+1})$. Note that we still have the property that for every $1 \leq i \leq n$ there is a set $S_i \subseteq A$ with $a_i \in S_i$ such that $(S_i, a_{i+1}) \in R$. We are going to discuss cycles in detail in Chapter 3.

Another notion we are going to need is the *primal graph*. There is plenty of knowledge about directed graphs, whereas directed hypergraphs have yet not been considered



(a)



(b)

Figure 2.2: (a) a SETAF and (b) its primal graph

very much in research. The primal graph of a SETAF can be used to define certain properties, we will later on conveniently use this to generalize the basic notions of acyclicity, symmetry, and bipartiteness (that are well studied for directed graphs) for directed hypergraphs, in particular for SETAFs. For an illustration of the primal graph see Figure 2.2.

Definition 10. *Let* $SF = (A, R)$ *be a SETAF. We define its* primal graph *as* $primal(SF) = (A', R')$, *where*

$$\begin{aligned} A' &= A, \\ R' &= \{(t, h) \mid (T, h) \in R, t \in T\}. \end{aligned}$$

Note that for AFs F we have $F = primal(F)$.

2.3 Complexity of Argumentation Frameworks

When it comes to reasoning in Argumentation Frameworks, several decision problems reflecting various degrees of scepticism have been proposed. The most well studied are *credulous* and *skeptical* reasoning. In general, we have that a semantics σ yields several extensions, yet, we are sometimes interested in whether to accept a single argument. The reasoning problems $Cred_\sigma$ and $Skept_\sigma$ are concerned with this very question. Whenever we will speak of "reasoning" throughout the course of this thesis, we will refer to these two problems.

Definition 11. *Let* σ *be a semantics, we define the following decision problems:*

- Credulous acceptance: *the problem* $Cred_\sigma$ *for a given SETAF* $SF = (A, R)$ *and a given argument* $a \in A$ *is to decide whether there is an extension* $E \in \sigma(SF)$ *such that* $a \in E$.
- Skeptical acceptance: *the problem* $Skept_\sigma$ *for a given SETAF* $SF = (A, R)$ *and a given argument* $a \in A$ *is to decide whether for every extension* $E \in \sigma(SF)$ *we have* $a \in E$.

Another problem we will occasionally encounter is the *verification problem*, that is, given a subset S of arguments of a SETAF SF, is S an extension of SF w.r.t. a semantics σ. Even though we will not explicitly state the complexity of verification for every sub-class, it will sometimes help us to come up with the results for the reasoning problems.

Definition 12. *Let σ be a semantics, we define the following decision problem:*

- Verification: *the problem Ver_σ for a given SETAF $SF = (A, R)$ and a given set of arguments $S \subseteq A$ is to decide whether S is an extension of SF w.r.t. σ, i.e. whether $S \in \sigma(SF)$.*

In the following, when we deal with the complexity of a decision problem for SETAFs, we assume that it is given in a *suitable representation* and taken as the input of a Turing machine. For AFs we have that the number of attacks is not significantly larger than the number of arguments (an AF with n arguments can have at most n^2 attacks), but for SETAFs the number of attacks can be exponential in the number of arguments, even if it is redundancy-free. Therefore for SETAFs we sometimes argue in terms of the number of attacks when it comes to the size of the input, whereas in AFs we sometimes take the number of arguments.

In Table 2.2 we see the complexity landscape for SETAFs (cf. [DGW18]). The complexity results coincide with the ones for AFs [DD17]. Throughout the thesis we will denote completeness for the complexity class $\mathcal{C}$ by $\mathcal{C}$-c.

If we can reduce a $\mathcal{C}$-hard problem to a SETAF decision problem such that the SETAF SF obtained by the reduction always has a syntactical property (such as acyclicity, symmetry, bipartiteness, etc.) we know that the problem is hard even for SETAFs with this property. In particular, this means we will not find an algorithm to solve the problem more efficiently than the respective complexity class suggests. We call SETAFs that share syntactic properties a *sub-class*, or, if the syntactic property regards the hypergraph structure of the SETAF, a *graph-class*.

If we know the complexity of reasoning problems of a sub-class to be lower than the general complexity (in particular, in P), and moreover we know we can decide the property efficiently (i.e. given a SETAF SF there is a deterministic Turing machine such that SETAFs with this property are the "yes"-instances), we speak of *tractable fragments*. We can potentially exploit the lower complexity of these fragments in the reasoning process by using efficient algorithms that cannot be applied in the general case, but yield the

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	P-c	P-c	Π_2^P-c	Π_2^P-c

Table 2.2: Complexity for AFs and SETAFs.

correct result if the input SETAF has the respective property. Tractable fragments for AFs are e.g. acyclic AFs, symmetric AFs, and bipartite AFs. In the respective chapters we will examine whether there are generalizations of these properties for SETAFs that form tractable fragments as well.

As hardness results are often obtained by similarly constructed reductions, we are now going to have a look at two of them. Note that we often show the hardness of reasoning in a specific sub-class for AFs, as these results carry over to SETAFs that generalize the sub-class. The first construction Reduction 1 will be referred to as the "standard reduction"; many of the hardness-results on the first level of the polynomial hierarchy can be obtained by applying this construction (or slightly changed variations thereof). Reduction 1 is illustrated in Figure 2.3.

Reduction 1. *Let φ be a CNF-formula consisting of a set of clauses C over a set of propositional atoms X. We define the SETAF $SF_1^\varphi = (A, R)$, where*

$$\begin{aligned} A &= \{\varphi\} \cup C \cup X \cup \bar{X}, \\ R &= \{(c, \varphi) \mid c \in C\} \ \cup \\ &\quad \{(x, c) \mid x \in c, c \in C\} \cup \{(\bar{x}, c) \mid \bar{x} \in c, c \in C\} \ \cup \\ &\quad \{(x, \bar{x}), (\bar{x}, x) \mid x \in X\} \end{aligned}$$

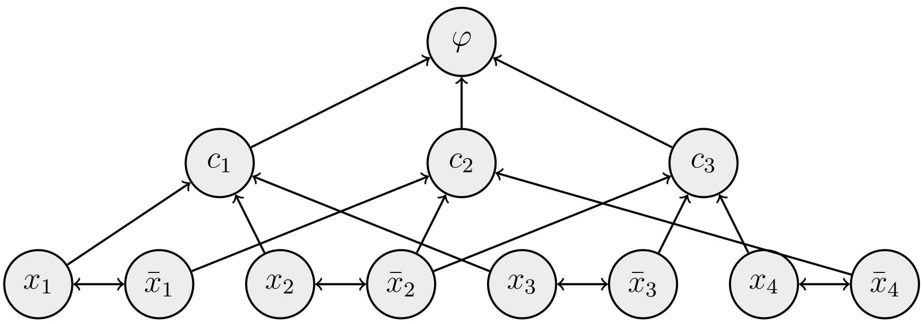


Figure 2.3: Illustration of the 'standard reduction' SF_1^φ for a formula φ with $X = \{x_1, x_2, x_3, x_4\}$, and $C = \{\{x_1, x_2, x_3\}, \{\bar{x}_1, \bar{x}_2, \bar{x}_4)\}, \{\bar{x}_2, \bar{x}_3, x_4\}\}$.

For Reduction 1 we have that φ is credulously accepted w.r.t. $\sigma \in \{adm, pref, stb, com\}$ iff the respective CNF-formula φ is satisfiable (cf. [DD17]).

We are also going to have a look at a variation of the standard reduction, as we are going to refer to this reduction multiple times in the course of this thesis. Reduction 2 from [Dvo12a] shows the Σ_2^P-hardness of $Cred_{stg}$ and the Π_2^P-hardness of $Skept_{stg}$. As it only contains a single cycle it is of particular interest for us. Reduction 2 is illustrated in Figure 2.4.

Reduction 2. *Let φ be a CNF-formula consisting of a set of clauses C over a set of propositional atoms X and let $x_\alpha \in X$ be an atom. Moreover let $\leq$ be an arbitrary total order on the clauses $c \in C$. We define the SETAF $SF_2^{\varphi,x_\alpha} = (A, R)$, where*

$$\begin{aligned}
A &= \{\varphi, b, q\} \cup C \cup X \cup \bar{X} \cup \{E_c \mid c \in C\}, \\
R &= \{(c, \varphi) \mid c \in C\} \cup \{(\varphi, b), (b, b), (q, x_\alpha)\} \cup \\
&\quad \{(\bar{x}, x) \mid x \in X\} \cup \\
&\quad \{(x, c) \mid x \in c, c \in C\} \cup \{(\bar{x}, c) \mid \bar{x} \in c, c \in C\} \cup \\
&\quad \{(E_c, a) \mid c \in C, a \in A \setminus (\{c, \varphi, b, q\} \cup \{E_{c'} \mid c' \leq c\})\}
\end{aligned}$$

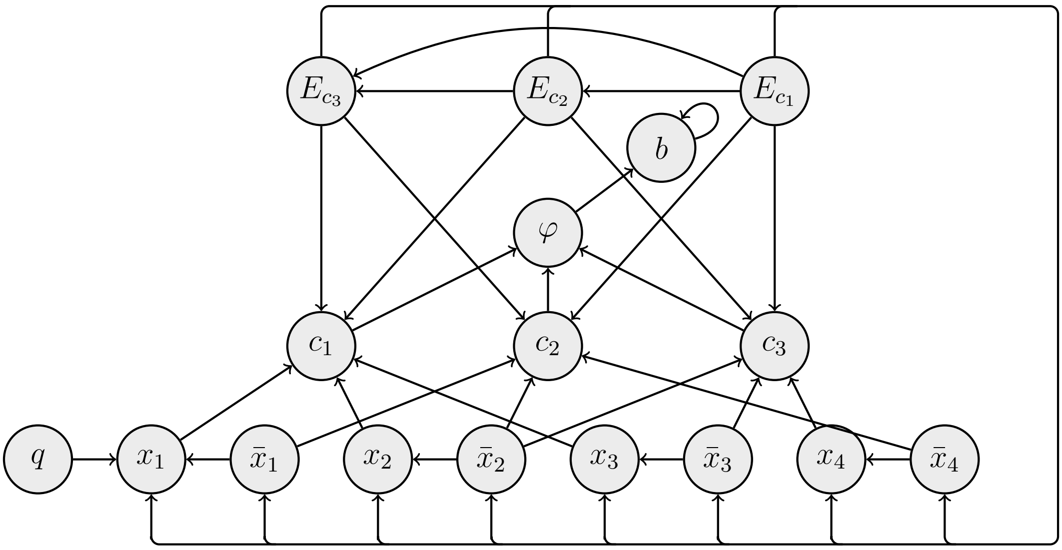


Figure 2.4: Illustration of SF_2^{φ,x_α} for a formula φ with $X = \{x_1, x_2, x_3, x_4\}$, and $C = \{\{x_1, x_2, x_3\}, \{\bar{x}_1, \bar{x}_2, \bar{x}_4\}, \{\bar{x}_2, \bar{x}_3, x_4\}\}$ and with $x_\alpha = x_1$. This illustrates, that reasoning problems for *stg* semantics are on the second level of the polynomial hierarchy, even for AFs with only one cycle of length 1.

For Reduction 2 we have that x_α is in a $\subseteq$-minimal model of φ iff x_α is credulously accepted iff q is not skeptically accepted w.r.t. *stg* semantics (cf. [Dvo12a]), showing that $Cred_{stg}$ and $Skept_{stg}$ are Σ_2^P-hard and Π_2^P-hard respectively, even for (SET)AFs with only one cycle of length 1.

Translations (cf. [DW11]) are a special kind of reduction, they allow us to easily reduce decision problems on SETAFs to other decision problems on SETAFs.

A (SETAF-)translation Tr is a function that takes a SETAF SF and outputs another SETAF $Tr(SF)$ with specific properties; if every SETAF in the range of the translation shares a syntactic property and certain semantic restrictions (i.e. certain relationships between the extensions apply) are preserved, we obtain a hardness-result for the class that is defined by the syntactic property. First we will adapt the syntactic notions for translations that are originally defined for AFs (see [DW11]) for SETAFs.

Definition 13. *A* (SETAF-)translation Tr *is a function which maps SETAFs to SETAFs. A translation* Tr *is called*

- efficient *if for every SETAF* SF *the SETAF* $Tr(SF)$ *can be computed efficiently,*
- embedding *if for every SETAF* SF *we have* $A^{SF} \subseteq A^{Tr(SF)}$ *and* $R^{SF} = R^{Tr(SF)} \cap ((2^{A^{SF}} \setminus \emptyset) \times A^{SF})$.

Now we will have a look at the semantic properties of translations. In particular, we will define three semantic restrictions, all of which are suitable for this kind of reduction. Again, the notions of exact and faithful translations are adapted from the respective definitions for AFs and similar formalisms [DW11, Jan99, Lib14], the notion of acceptance-preserving translations is a weaker form thereof, which still suffices for our purposes.

Definition 14. *Let* σ, σ' *be semantics, then a (SETAF-)translation* Tr *is called*

- exact *for* $\sigma \Rightarrow \sigma'$ *if for every SETAF* SF *we have* $\sigma(SF) = \sigma'(Tr(SF))$,
- faithful *for* $\sigma \Rightarrow \sigma'$ *if for every SETAF* SF *we have* $\sigma(SF) = \{E \cap A^{SF} \mid E \in \sigma'(Tr(SF))\}$ *and* $|\sigma(SF)| = |\sigma'(Tr(SF))|$,
- acceptance-preserving *for* $\sigma \Rightarrow \sigma'$ *if for every SETAF* SF *we have* $\sigma(SF) = \{E \cap A^{SF} \mid E \in \sigma'(Tr(SF))\}$.

We have that every exact translation is faithful, and every faithful translation is acceptance-preserving. As we are often not so much interested in the translatability between different semantics, but in the syntactic properties of the translations, we often use translations where $\sigma = \sigma'$, then we just write "Tr is an exact/faithful/acceptance-preserving translation for σ".

We use acceptance-preserving translations as reductions in the following way: if a reduction Tr ensures that for every SETAF SF its translated pendant $Tr(SF)$ has a special syntactic property, then the hardness of both credulous and skeptical reasoning carry over from general SETAFs to SETAFs with this property.

CHAPTER 3

Cycles and Acyclicity

In this chapter we will investigate what effect the existence, parity, and length of cycles in a SETAF have on the complexity of reasoning problems. In particular, in Section 3.1 we will recapitulate the existing results for acyclic, even-cycle-free, and odd-cycle-free AFs that pose a lower bound for SETAFs. After that, in Section 3.2 we will adapt these notions for SETAFs and work out the complexity thereof.

In Section 3.3 we will investigate how restricting the length of cycles affects the complexity of reasoning in AFs and SETAFs. Finally, in Section 3.4 we will suggest adaptations from acyclicity-notions of hypergraphs for SETAFs and look at properties of these classes of SETAFs. Here we will utilize the results obtained while investigating the minimal and maximal cycle lengths.

3.1 State of the Art in AFs

For AFs we know that acyclicity, even-cycle-freeness, and odd-cycle-freeness are properties that allow us to reason more efficiently. In particular, the complexity of the reasoning problems w.r.t. the semantics under our consideration is given in Table 3.1. The results for acyclic AFs are from [Dvo12a], for even-cycle-free AFs from [DB01] and [Dvo12a], and for odd-cycle-free AFs from [Dun07] and [DD17][1].

The identified tractable fragments are acyclic AFs for all semantics under our consideration and even-cycle-free AFs for $\sigma \in \{cf, adm, naive, stb, pref, com, grd, sem\}$. Moreover some work has been done on the complexity of AFs that do not have self-loops. We will generalize this and investigate how a minimal cycle length of k affects the complexity

[1] There is a typing error in Table 4 of the cited paper regarding the complexity results for odd-cycle free AFs. $Skept_{stb}$ is coNP-complete and $Skept_{com}$ is P-complete for odd-cycle-free AFs.

Acyclic AFs

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	P-c	P-c

Even-cycle-free AFs

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	Σ_2^P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	Π_2^P-c	P-c

Odd-cycle-free AFs

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	NP-c	NP-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	coNP-c	P-c	P-c	coNP-c	coNP-c

Table 3.1: Complexity for acyclic, even-cycle-free, and odd-cycle-free AFs.

(where k is a constant). Odd-cycle-free AFs are still intractable, but the complexity drops to the first level of the polynomial hierarchy.

3.2 Acyclicity in SETAFs

In this section we will adapt the notions discussed in the previous section that have been investigated for AFs in a natural way for SETAFs and determine the respective complexity results.

3.2.1 Cycle-free SETAFs

As we need the notion of cycles in all chapters, it is already defined in the preliminaries (see Definition 9). We have that a SETAF SF is acyclic (in the SETAF sense) iff its primal graph $primal(SF)$ is acyclic (in the classical sense of directed graphs). To distinguish this notion of acyclicity from others, we sometimes denote it by 'classical' acyclicity.

Definition 15. *A SETAF is* acyclic *if it contains no cycles, otherwise it is* cyclic*.*

The structure of acyclic SETAFs is especially beneficial when one is computing the grounded extension. In particular, we have that the grounded extension of acyclic SETAFs is stable, as the next lemma shows.

Lemma 1 (cf. [DD17, p. 657])**.** *Let $SF = (A, R)$ be an acyclic SETAF and let G be the grounded extension of SF. Then each argument $a \in A$ is either in G or attacked by G.*

Proof. Assume a non-empty set $A' \subseteq A$ such that $G \cap A' = \emptyset$ and there is no argument $a \in A'$ that is attacked by G. Since SF is acyclic, the SETAF $SF_{A'} = (A', \{(S, a) \mid a \in$

$A', S \subseteq A', (S,a) \in R\}$) (i.e. the SETAF resulting from restricting SF to the arguments in A') is acyclic as well. Therefore there is an argument $b \in A'$ that is defended by G, so $b \in G$, which contradicts the assumption. □

Moreover, the very restrictive nature of acyclicity is highlighted by the fact that every acyclic SETAF is well-founded (see [NP06, p. 65]). Well-founded SETAFs are generally easy to reason on, but many problems cannot be expressed with well-founded SETAFs.

Lemma 2. *Let $SF = (A, R)$ be an acyclic SETAF. Then SF is well-founded.*

Proof. Since A is finite and SF is acyclic, it is obvious that there cannot be an infinite sequence of sets $B_1, B_2, \ldots,$ such that for all i, B_i is the tail of an attack towards an argument in B_{i-1}. □

Lemma 3. *Let $SF = (A, R)$ be an acyclic SETAF. Then $grd(SF) = com(SF) = pref(SF) = stb(SF) = sem(SF) = stg(SF)$.*

Proof. From Lemma 2 and the fact that for well-founded SETAFs the grounded, complete, preferred, and stable semantics coincide (see [NP06, p. 65]) we get that $grd(SF) = com(SF) = pref(SF) = stb(SF)$. There is a stable extension, hence the stable, semi-stable and stage semantics coincide, i.e. $stb(SF) = sem(SF) = stg(SF)$. □

These results immediately give us an upper bound for our reasoning problems over acyclic SETAFs. As all of the semantics mentioned in Lemma 3 coincide with the grounded semantics, we know that there is exactly one extension for the semantics. This also implies that the respective credulous and skeptical reasoning problem coincide, as an argument is in one extension iff it is in every extension.

Lemma 4. *Let $\sigma \in \{adm, grd, com, pref, stb, sem, stg\}$. Then for acyclic SETAFs the problems $Skept_\sigma$, $Cred_\sigma$, and Ver_σ are in P.*

Proof. The membership for admissible semantics follows from the fact that an argument is credulously accepted w.r.t. admissible semantics in an acyclic SETAF iff it is in the grounded extension. From Lemma 3 we get that the problems coincide with the respective problems for the grounded semantics, so the general results for the grounded semantics in SETAFs apply (see [DGW18, p. 13]). □

Lemma 5. *Let $\sigma \in \{adm, grd, com, pref, stb, sem, stg\}$. Then for acyclic SETAFs the problems $Skept_\sigma$ and $Cred_\sigma$ are P-hard.*

Proof. This holds even for acyclic AFs, which are special cases of acyclic SETAFs (see Table 3.1). □

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	P-c	P-c

Table 3.2: Complexity for acyclic SETAFs (coincides with acyclic AFs).

The last two results suffice to pin down the complexity of reasoning on acyclic SETAFs w.r.t. the semantics in question.

Theorem 1. *For acyclic SETAFs the complexity results as given in Table 3.2 hold.*

Proof. From Lemma 4 and Lemma 5 and the results for general SETAFs. □

To further state that acyclic SETAFs form a *tractable fragment* we also need to establish that deciding whether a SETAF is acyclic is tractable as well. As we already know a SETAF is acyclic iff its primal graph is acyclic, this reduces to check acyclicity for a directed graph. It is well known that this problem is efficiently decidable, one general algorithm to do so is to remove leafs (nodes with no outgoing edges) until every node is removed. If this is possible, the graph is acyclic, if not, it contains a cycle. Together with Theorem 1 this means we identified a tractable fragment in our sense.

Theorem 2. *Acyclicity is a tractable fragment for SETAFs for all semantics under our consideration.*

3.2.2 Even-cycle-free SETAFs

The complexity results for even-cycle-free AFs also apply for SETAFs, the proofs only need minor adaptations. The following is analogous to the proofs given in [Dvo12a, p. 80-81].

Lemma 6 (cf. [Dvo12a, p. 80])**.** *Let $SF = (A, R)$ be a SETAF. If $|com(SF)| \geq 2$ then SF contains an even-cycle.*

Proof. Let G be the grounded extension of SF and E a complete extension such that $E \neq G$. We thus have $E \supset G$. This means there is some $x_0 \in E \setminus G$ with $(Y_0, x_0) \in R$ such that $G \not\mapsto^R Y_0$. Since E is conflict-free and we have $x_0 \in E$, we also have $Y_0 \not\subseteq E$, i.e. there is an argument $y \in Y_0$ such that $y \notin E$. In order to defend x_0 in E there is some set $X_1 \subseteq E$ such that $(X_1, y_0) \in R$ for some $y_0 \in Y_0 \setminus E$. Now the same reasoning holds for some $x_1 \in X_1 \setminus G$, and inductively we get an infinite sequence $x_0, y_0, x_1, y_1, \dots$ such that $x_i \in E \setminus G$ and $(Y_i, x_i), (X_{i+1}, y_i) \in R$ with $y_i \in Y_i$ and $x_i \in X_i$ for $i \geq 1$. Since $E \setminus G$ is finite we get that $x_i = x_j$ for some $i \neq j$, then $(x_j, (X_j, y_{j-1}), y_{j-1}, (Y_{j-1}, x_{j-1}), x_{j-1}, \dots, (Y_i, x_i), x_i)$ is an even-cycle. □

Theorem 3. *For even-cycle-free SETAFs the complexity results as given in Table 3.3 hold.*

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	Σ_2^P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	Π_2^P-c	P-c

Table 3.3: Complexity for even-cycle-free SETAFs (coincides with even-cycle-free AFs).

Proof. The hardness follows from the hardness of the respective problems for AFs (see (see Table 3.1). By Lemma 6 we have that if a SETAF does not contain an even-cycle then it only has a single complete extension. As all grounded, preferred, and semi-stable extensions are complete and yield at least one extension, these semantics coincide with the complete semantics. As also every stable extension is complete there is only one candidate for stable extensions, so it suffices to check if the grounded extension is stable, which can be done in polynomial time. If the grounded extension is stable, $Cred/Skept_{stb}$ coincide with $Cred/Skept_{grd}$, otherwise they are trivially false. The P-membership of $Cred_{adm}$ follows from the identity $Cred_{adm} = Cred_{com}$. The membership for the stage semantics follows from the general case. □

As it is efficiently decidable whether a given directed graph has an even cycle [McC04], and a SETAF has an even cycle iff its primal graph has an even cycle, even-cycle-free SETAFs form a tractable fragment for all semantics under our consideration except for stage.

Theorem 4. *Even-cycle-freeness is a tractable fragment for SETAFs for* $\sigma \in \{cf, adm, naive, stb, pref, com, grd, sem\}$.

3.2.3 Odd-cycle-free SETAFs

As for AFs we have that odd-cycle-free SETAFs are limited controversial and therefore coherent. These notions from [Dun95, p. 331-333] were adapted for SETAFs in [NP06, p. 66]. For coherent (SET)AFs all preferred extensions are stable, together with the results for odd-cycle free AFs (see [DD17, p. 28]) we get the results for odd-cycle free SETAFs.

Lemma 7. *Odd-cycle-free SETAFs are limited controversial.*

Proof. Assume towards contradiction there is an odd-cycle-free SETAF SF that is not limited controversial, i.e. with an infinite sequence of arguments $a_1, a_2, \ldots$ such that for all $i \geq 1$ the argument a_i is controversial with respect to a_{i-1}. Since A^{SF} is finite, we have that $a_i = a_j$ for some $j > i$. This means there is a cycle from a_i to a_j. Furthermore, since a_i indirectly attacks and indirectly defends a_j there must be a cycle of odd length from a_i to a_j, which is a contradiction to the assumption that SF was odd-cycle-free. □

This result is already sufficient for our result about odd-cycle-free SETAFs, that is, the complexity of odd-cycle-free SETAFs coincides with the complexity of odd-cycle-free AFs.

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	NP-c	NP-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	coNP-c	P-c	P-c	coNP-c	coNP-c

Table 3.4: Complexity for odd-cycle-free SETAFs (coincides with odd-cycle-free AFs).

Theorem 5. *For odd-cycle-free SETAFs the complexity results as given in Table 3.4 hold.*

Proof. From the results for AFs we get a lower bound for the complexity (see Table 3.1). From Lemma 7 we get that odd-cycle-free SETAFs are limited controversial and therefore coherent (see [NP06, p. 68]), hence by definition all of its preferred extensions are stable, which means there is at least one stable extension. Now we have that the semi-stable and stage extensions coincide with the stable extensions. □

Even though the complexity for some semantics is lower than in the general case, most reasoning tasks are still intractable, which means we cannot speak of a tractable fragment for SETAFs.

3.3 Cycle Length

As many notions of acyclicity that are to be discussed restrict cycles to be of certain minimal or maximal lengths, an examination of the effects of such restrictions on the complexity of the respective decision problems is needed. In the following we will define and investigate SETAFs with these properties and will also explicitly state the results for AFs.

3.3.1 Minimal Cycle Length

In this section we will investigate how restricting the minimal cycle length of AFs and SETAFs affect the complexity of our reasoning problems. One often deals with (SET)AFs that have 'no short cycles', i.e. it is guaranteed that every cycle has at least k arguments for some fixed $k \geq 0$.

Definition 16. *Let SF be a SETAF. Then its* minimal cycle length $mincl(SF)$ *is defined as the length of the shortest cycle in SF, and 0 if SF is acyclic.*

Our main goal here is to determine the complexity for SETAFs that have no cycles that are shorter than k arguments. As for $k = 0$ this problem coincides with the classical decision problems, we will examine whether this is also true for $k = 1, 2, \ldots$.
To this end we will prove a stronger result, that is, for $\sigma \in \{cf, adm, naive, stb, pref, com, grd, stg, sem\}$ we provide translations Tr for an arbitrary SETAF SF to a SETAF

$Tr(SF)$ with $mincl(Tr(SF)) \geq k$, such that the decision problems under our consideration ($Cred_\sigma$ and $Skept_\sigma$) coincide, i.e. such that Tr is acceptance-preserving. Moreover for $\sigma \in \{cf, naive, stb, pref, com, grd, sem\}$ we will also show that the provided translations are faithful, i.e. also $|\sigma(SF)| = |\sigma(Tr(SF))|$ holds.

We first present a translation for *grd*, *stb*, *com*, *pref*, and *sem* semantics such that for any SETAF SF its translation $Tr(SF)$ has no cycles shorter than k arguments for an arbitrarily large constant k. For an illustration of Tr_1^k see Figure 3.1.

Translation 1. *The SETAF-translation Tr_1^k for $k \geq 1$ is defined as $Tr_1^k(SF) = (A', R')$, where*

$$\begin{aligned} A' &= A^{SF} \cup \{a_1^{(T,h)}, \dots, a_{2k}^{(T,h)} \mid (T,h) \in R^{SF}\} \\ R' &= \{(T, a_1^{(T,h)}), (a_1^{(T,h)}, a_2^{(T,h)}), \dots, (a_{2k-1}^{(T,h)}, a_{2k}^{(T,h)}), (a_{2k}^{(T,h)}, h) \mid (T,h) \in R^{SF}\} \end{aligned}$$

Intuitively, Tr_1^k 'extends' every attack, such that an original attack corresponds to $2k+1$ consecutive attacks. For semantics based on admissible sets with a maximality constraint (such as *grd*, *stb*, *com*, *pref*, and *sem*) this is a faithful translation (as we will show) to a SETAF that does not contain any cycles that are shorter than $k/2$.
If for some extension E and some attack (T,h) we have that $T \subseteq E$, in the corresponding extension E' of the translation we have that for this attack all arguments $a_i^{(T,h)}$ with an even i will be in E'. If T is attacked on the other hand, then $a_j^{(T,h)}$ for odd j will be defended, and those arguments will be in E'. If neither $T \subseteq E$ nor $E \rightarrowtail T$, then no additional argument will be in E' for this attack. Formally, this is captured in the *corresponding extension* for the translation.

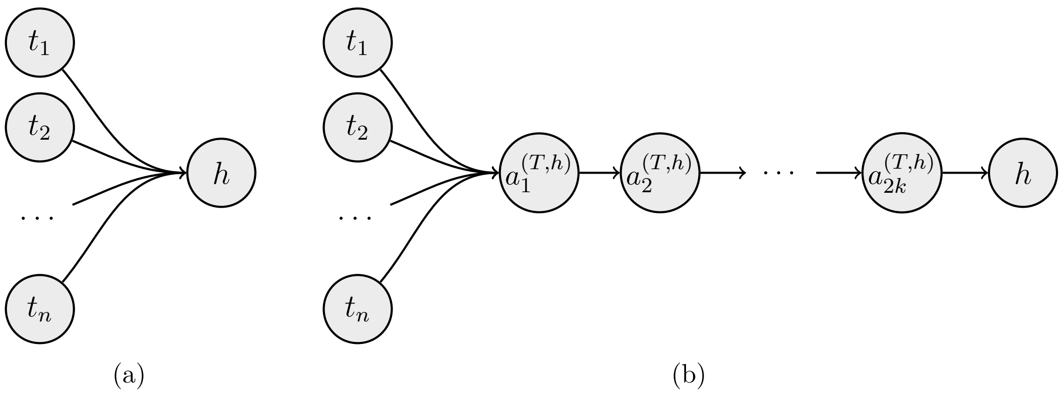


Figure 3.1: Illustration of translation Tr_1^k. We see a SETAF with one attack (T,h) with $T = \{t_1, t_2, \dots, t_n\}$ before (a) and after (b) the translation.

Definition 17. *Let $SF = (A, R)$ be a SETAF, $E \subseteq A$, and $k \geq 1$. Then*

$$\begin{aligned} M^{SF}_{Tr^k_1}(E) \;=\; & E \;\cup \\ & \{a^{(T,h)}_{2i} \mid 1 \leq i \leq k, (T,h) \in R, T \subseteq E\} \;\cup \\ & \{a^{(T,h)}_{2i-1} \mid 1 \leq i \leq k, (T,h) \in R, E \rightarrowtail^R T\} \end{aligned}$$

is the corresponding extension *of E for the translation $Tr^k_1(SF)$.*

In order to show the faithfulness of Tr^k_1 we will first establish some properties of the translation. For 'original' arguments, i.e. arguments from A, we have that they are attacked by an extension in the translation iff they are attacked by the corresponding extension in the translation.

Lemma 8. *Let $SF = (A, R)$ be a SETAF, $E \subseteq A$, $k \geq 1$, $SF' = Tr^k_1(SF)$, and $E' = M^{SF}_{Tr^k_1}(E)$. Then for every argument $a \in A$ we have that $E \rightarrowtail^R a$ iff $E' \rightarrowtail^{R^{SF'}} a$.*

Proof. "⇒": $E \rightarrowtail^R a$ means for every attack $(T, a) \in R$ that $T \subseteq E$, so $T \subseteq E'$, which by construction means $a^{(T,a)}_{2k} \in E'$ for every attack towards a, i.e. a is attacked by E' in SF'.
"⇐": $E' \rightarrowtail^{R^{SF'}} a$ means for every attack $(T, a) \in R$ that $a^{(T,a)}_{2k} \in E'$. By construction this is only the case if $T \subseteq E$ for every attack towards a, therefore $E \rightarrowtail^R a$. □

Now we will establish that the semantics *grd*, *adm*, *stb*, *com*, *pref*, and *sem* correspond in a SETAF SF and its translation $Tr^k_1(SF)$.

Lemma 9. *Let $SF = (A, R)$ be a SETAF, $E \subseteq A$, $k \geq 1$, $SF' = Tr^k_1(SF)$, and $E' = M^{SF}_{Tr^k_1}(E)$. Then $E \in grd(SF)$ iff $E' \in grd(SF')$.*

Proof. Let G be the grounded extension of SF and let G' be the grounded extension of SF'. Consider the construction of G and G' and via their respective characteristic functions $\mathcal{F}_{SF}$ and $\mathcal{F}_{SF'}$. Let $G_i = \mathcal{F}^i_{SF}(\emptyset)$, i.e. the set that results from i applications of the characteristic function on the empty set. Likewise, let $G'_i = \mathcal{F}^i_{SF'}(\emptyset)$. Since every attack was 'extended' by $2k$ arguments we have that one step in the construction of G corresponds to $2k + 1$ steps of the construction of G'. In particular we have that $G'_j = M^{SF}_{Tr^k_1}(G_i)$ where $j = i * (2k + 1)$ for $i \geq 0$.

□

Lemma 10. *Let $SF = (A, R)$ be a SETAF, $k \geq 1$, $SF' = Tr^k_1(SF)$. Then we have for any set $E \subseteq A$ that $E \in adm(SF)$ iff $E' = M^{SF}_{Tr^k_1}(E) \in adm(SF')$.*

Proof. "$\Rightarrow$": First note that E' is conflict-free. Moreover we have that every $a \in E'$ is defended in SF': if a is in E, i.e. a is an original argument, then every attacking set of a is attacked by E in SF. This means that for every attack $(T, a) \in R$ we have that $a_{2k-1}^{(T,a)}$ is in E', so a is defended by E' in SF'. If a is in $E' \setminus E$, i.e. a is a newly added argument, then a is of the form $a = a_i^{(T,h)}$ for some attack (T, h) and some $1 \leq i \leq 2k$. If $i \geq 3$, then $a_{i-2}^{(T,h)}$ is in E', so $a_i^{(T,h)}$ is defended. If $i = 2$ we have that $T \subseteq E$, so $a_2^{(T,h)}$ is defended. If $i = 1$ we have that $E \rightarrowtail^R T$, hence, $E' \rightarrowtail^{R^{SF'}} T$, so $a_1^{(T,h)}$ is defended.
"$\Leftarrow$": We show the stronger result that if E' is admissible in SF', then $E = E' \cap A$ is admissible in SF. Towards contradiction assume the contrary. This means either E is not conflict-free or some argument $a \in E$ is not defended by E in SF. Assume E is not conflict-free, i.e. for some attack $(T, h) \in R$ we have $(T \cup \{h\}) \subseteq E$. Since $E \subseteq E'$ we also have $(T \cup \{h\}) \subseteq E'$. Since h is defended by E' in SF' we have $\{a_{2i-1}^{(T,h)} \mid 1 \leq i \leq k\} \subseteq E'$, but since $T \subseteq E'$ and $a_1^{(T,h)} \in E'$ we have that E' is not conflict-free, which is a contradiction to the assumption that E' is admissible. Now assume some argument $a \in E$ is not defended by E in SF, i.e. there is an attack $(T, a) \in R$ such that $E \not\rightarrowtail^R T$. Since $a \in E$ we have that $a \in E'$, in order to defend it in SF' we have $\{a_{2i-1}^{(T,a)} \mid 1 \leq i \leq k\} \subseteq E'$ for every attack (T, a) towards a. To defend $a_1^{(T,a)}$ in SF' we have $E' \rightarrowtail^{R^{SF'}} T$, so there is an argument $t \in T$ that is attacked by E', in particular by some argument $a_{2k}^{(S,t)}$, where $(S, t) \in R$. To defend $a_{2k}^{(S,t)}$ we have that $\{a_{2i}^{(S,t)} \mid 1 \leq i \leq k\} \subseteq E'$, and to defend $a_2^{(S,t)}$ we have $S \subseteq E'$, so $S \subseteq E$, hence, a is indeed defended by E in SF, which is a contradiction. □

Lemma 11. *Let $SF = (A, R)$ be a SETAF, $k \geq 1$, $SF' = Tr_1^k(SF)$. Then $adm(SF) = \{E \cap A^{SF} \mid E \in adm(Tr(SF))\}$.*

Proof. The "$\subseteq$"-direction follows from Lemma 10. For the "$\supseteq$"-direction recall that in the proof of Lemma 10 we already showed that if E' is admissible in SF' then $E = E' \cap A$ is admissible in SF. □

For *adm* semantics we have additional extensions in the translation, as for every 'extended' attack a set can be admissible even if not all of the added arguments a_{2i} (or a_{2i+1} respectively) are in it. As the semantics *stb*, *com*, *pref*, and *sem* all require maximality of a certain kind, we have that the extensions in the translation are all of the same form, as the next lemma shows.

Lemma 12. *Let $SF = (A, R)$ be a SETAF, $k \geq 1$, and let $SF' = Tr_1^k(SF)$. Then for $\sigma \in \{stb, com, pref, sem\}$ we have that every extension $E' \in \sigma(SF')$ is of the form $E' = M_{Tr_1^k}^{SF}(E)$ for some $E \subseteq A$.*

Proof. Towards contradiction assume there is an extension $E' \in \sigma(SF')$ that is not of this form, and let $E = E' \cap A$. This happens in four cases:

(a) There is some $a_{2i}^{(T,h)} \in E'$ where $1 \leq i \leq k$ with $T \not\subseteq E$. In order to defend $a_{2i}^{(T,h)}$ we have $a_{2j}^{(T,h)} \in E'$ with $1 \leq j \leq i$, but $a_2^{(T,h)}$ is not defended, which is a contradiction to the assumption that $E' \in \sigma(SF)$.

(b) There is some $a_{2i}^{(T,h)} \notin E'$ where $1 \leq i \leq k$, but we have $T \subseteq E$. Since it is defended we have $a_2^{(T,h)} \in E'$, likewise we have $a_{2j}^{(T,h)} \in E'$ for $1 \leq i \leq k$, which is a contradiction.

(c) There is some $a_{2i-1}^{(T,h)} \in E'$ where $1 \leq i \leq k$ with $E \not\rightarrowtail^R T$. In order to defend $a_{2i-1}^{(T,h)}$ we have $a_{2j-1}^{(T,h)} \in E'$ with $1 \leq j \leq i$, but $a_1^{(T,h)}$ is not defended, which is a contradiction to the assumption that $E' \in \sigma(SF)$.

(d) There is some $a_{2i-1}^{(T,h)} \notin E'$ where $1 \leq i \leq k$, but we have $E \rightarrowtail^R T$. Since it is defended we have $a_1^{(T,h)} \in E'$, likewise we have $a_{2j-1}^{(T,h)} \in E'$ for $1 \leq i \leq k$, which is a contradiction.

Since all possible ways E' could be of other form lead to a contradiction, E' must be of the desired form. □

Lemma 13. *Let $SF = (A, R)$ be a SETAF, $k \geq 1$, $SF' = Tr_1^k(SF)$. Then $stb(SF') = \{M_{Tr_1^k}^{SF}(E) \mid E \in stb(SF)\}$.*

Proof. It is sufficient to show that $E \in stb(SF)$ iff $E' = M_{Tr_1^k}^{SF}(E) \in stb(SF')$, then together with Lemma 12 the claim follows.
"⇒": Since E is stable, we have that for every attack $(T, h) \in R$ either $T \subseteq E$ or $E \rightarrowtail^R T$. If for some attack (T, h) we have $T \subseteq E$, then we have $\{a_{2i}^{(T,h)} \mid 1 \leq i \leq k\} \subseteq E'$, especially we have that $a_{2k}^{(T,h)} \in E'$, so $E' \rightarrowtail^{R^{SF'}} h$ and $E' \rightarrowtail^{R^{SF'}} a_{2i-1}^{(T,h)}$ for $1 \leq i \leq k$. Similarly, if for some $(T, h) \in R$ we have $E \rightarrowtail T$, then we have $\{a_{2i-1}^{(T,h)} \mid 1 \leq i \leq k\} \subseteq E'$ and $E' \rightarrowtail a_{2i}^{(T,h)}$ for $1 \leq i \leq k$.
"⇐": observe that for any attack $(T, h) \in R$ by construction we have that h is attacked by E' in SF' iff $T \subseteq E'$ which is the case iff $T \subseteq E$, so every argument $a \in A$ has $E \rightarrowtail^R a$ iff $a \notin E$.

□

Lemma 14. *Let $SF = (A, R)$ be a SETAF, $k \geq 1$, $SF' = Tr_1^k(SF)$. Then $com(SF') = \{M_{Tr_1^k}^{SF}(E) \mid E \in com(SF)\}$.*

Proof. It is sufficient to show that $E \in com(SF)$ iff $E' = M_{Tr_1^k}^{SF}(E) \in com(SF')$, then together with Lemma 12 the claim follows.
"⇒": By Lemma 10 we have that E' is admissible. It remains to show that no argument $a \in A^{SF'} \setminus E'$ is defended by E' in SF'. Towards contradiction assume there is such an

argument a that is defended by E'm but not in E', which leads to three cases: $a \in A$, $a = a_{2i-1}^{(T,h)}$, or $a = a_{2i}^{(T,h)}$ for some $1 \leq i \leq k$ and some attack $(T,h) \in R$. Assume $a \in A$. Since it is defended by E' in SF', we have that for every attack $(T,a) \in R$ the arguments $a_{2k-1}^{(T,a)}$ are in E'. Hence, by construction, $E \rightarrowtail^R T$ for every attack (T,a) towards a. But then a is defended by E in SF without being in E, which is a contradiction to the assumption that E is complete. Assume $a = a_{2i-1}^{(T,h)}$. Since $a \notin E'$, if $i \geq 2$ then by construction we have $a_{2i-3}^{(T,h)} \notin E'$, so a is not defended by E', which is a contradiction. If $i = 1$, then by construction $E \not\rightarrowtail^R T$, so by Lemma 8 $E' \not\rightarrowtail^{R^{SF'}} T$, so again a is not defended by E', which is a contradiction. Finally, assume $a = a_{2i}^{(T,h)}$. Again, if $i \geq 2$ then by construction we have $a_{2i-2}^{(T,h)} \notin E'$, so a is not defended by E', which is a contradiction. If $i = 1$, then by construction $T \not\subseteq E$, so $T \not\subseteq E'$, therefore a is not defended by E', which is a contradiction.
"$\Leftarrow$": Again, by Lemma 10 we have that E is admissible. It remains to show that no argument $a \in A \setminus E$ is defended by E in SF. Towards contradiction assume there is such an argument a that is defended by E, but not in E. This means for every attack $(T,a) \in R$ towards a that $E \rightarrowtail^R T$, but then we would have that for these attacks $a_{2k-1}^{(T,a)} \in E'$, so a is defended by E' in SF', which is a contradiction to the assumption that E' is complete. □

Lemma 15. *Let $SF = (A,R)$ be a SETAF, $k \geq 1$, $SF' = Tr_1^k(SF)$. Then $pref(SF') = \{M_{Tr_1^k}^{SF}(E) \mid E \in pref(SF)\}$.*

Proof. It is sufficient to show that $E \in pref(SF)$ iff $E' = M_{Tr_1^k}^{SF}(E) \in pref(SF')$, then together with Lemma 12 the claim follows.
"$\Rightarrow$": As every preferred extension is complete, by Lemma 14 we have that E' is complete. It remains to show that there is no $T' \supset E'$ such that $T' \in adm(SF')$. Towards contradiction assume there is such a set T' with $T' \setminus S' \neq \emptyset$. Then by Lemma 10 we have that $T = T' \cap A$ is admissible in SF with $T \supset E$, which is a contradiction to the assumption that E is preferred.
"$\Leftarrow$": Again, by Lemma 14 we have that E is complete. It remains to show that there is no $T \supset E$ such that $T \in adm(SF)$. Towards contradiction assume there is such a set T with $T \setminus S \neq \emptyset$. Then by Lemma 10 we have that $T' = M_{Tr_1^k}^{SF}$ is admissible in SF' with $T' \supset E'$, which is a contradiction to the assumption that E' is preferred.

□

Lemma 16. *Let $SF = (A,R)$ be a SETAF, $k \geq 1$, $SF' = Tr_1^k(SF)$ with $SF' = (A',R')$. Then $sem(SF') = \{M_{Tr_1^k}^{SF}(E) \mid E \in sem(SF)\}$.*

Proof. It is sufficient to show that $E \in sem(SF)$ iff $E' = M_{Tr_1^k}^{SF}(E) \in sem(SF')$, then together with Lemma 12 the claim follows.
"$\Rightarrow$": As every semi-stable extension is preferred, by Lemma 15 we have that E' is

preferred. It remains to show that there is no $T' \subseteq A^{SF'}$ such that $T' \in adm(SF')$ and $T'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$. Towards contradiction assume there is such a set T'. Then by Lemma 10 we have that $T = T' \cap A$ is admissible in SF with $T^{\oplus}_{R} \supset E^{\oplus}_{R}$, which is a contradiction to the assumption that E is semi-stable in SF.
"$\Leftarrow$": Again, by Lemma 15 we have that E is preferred. It remains to show that there is no $T \subseteq A$ such that $T \in adm(SF)$ and $T^{\oplus}_{R} \supset E^{\oplus}_{R}$. Towards contradiction assume there is such a set T. Then by Lemma 10 we have that $T' = M^{SF}_{Tr^k_1}$ is admissible in SF' with $T'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$, which is a contradiction to the assumption that E' is semi-stable in SF'. □

These results allow us to formulate the following proposition and theorem regarding Tr^k_1.

Proposition 1. *For $\sigma = adm$ we have that Tr^k_1 is an efficient, acceptance-preserving translation for $\sigma \Rightarrow \sigma$ such that $mincl(Tr^k_1(SF)) \geq k$ for every SETAF SF and every $k \geq 1$.*

Proof. Immediate by Lemma 11. □

Theorem 6. *For $\sigma \in \{grd, stb, com, pref, sem\}$ we have that Tr^k_1 is an efficient, faithful translation for $\sigma \Rightarrow \sigma$ such that $mincl(Tr^k_1(SF)) \geq k$ for every SETAF SF and every $k \geq 1$.*

Proof. $Tr^k_1(SF)$ for some SETAF SF can be efficiently computed, as only $2k * |R^{SF}|$ arguments and $(2k+1) * |R^{SF}|$ attacks are added to SF.
$Tr^k_1(SF)$ being a faithful translation follows from Lemma 9, Lemma 13, Lemma 14, Lemma 15, and Lemma 16, and the fact that for a SETAF SF and two sets $S, S' \subseteq A^{SF}$ with $S \neq S'$ we have $M^{SF}_{Tr^k_1}(S) \neq M^{SF}_{Tr^k_1}(S')$. □

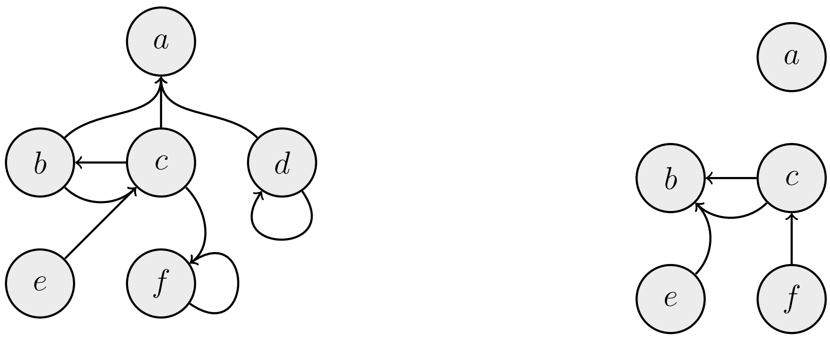
Corollary 1. *For $\sigma \in \{grd, stb, com, pref, sem\}$ we have that Tr^k_1 is an efficient, faithful translation for $\sigma \Rightarrow \sigma$ such that $mincl(Tr^k_1(F)) \geq k$ for every AF F and every $k \geq 1$.*

Proof. Immediate as Tr^k_1 maps AFs to AFs. □

Note that in general, Tr^k_1 does not work for semantics that are based on conflict-free sets, as it is not guaranteed that for an extended attack either all even or all odd arguments are in a conflict-free set. To this end, for *cf* and *naive* semantics we introduce a different translation. For an illustration of Tr_2 see Figure 3.2.

Translation 2. *Let $\leq$ be a total order on the arguments A of a SETAF SF. The SETAF-translation Tr_2 is defined as $Tr_2(SF) = (A', R')$, where*

$$\begin{aligned} A' &= A^{SF} \setminus \{a \mid (a,a) \in R^{SF}\} \\ R' &= \{(S \setminus \{x\}, x) \mid (T,h) \in R^{SF}, S = T \cup \{h\}, x \in S, \\ &\quad \forall y \in S \text{ we have } (y,y) \notin R^{SF} \text{ and } x \leq y\} \end{aligned}$$



(a) (b)

Figure 3.2: Illustration of translation Tr_2. We see a SETAF before (a) and after (b) the translation with $a \leq b \leq c \leq d \leq e \leq f$.

For conflict-free sets the direction of attacks does not play a role, so Tr_2 'reorients' the attacks towards the least argument w.r.t. $\leq$. As that are involved in a self-loop never appear in conflict-free sets, we can omit them. Moreover, if such an argument appears in the tail or head of an attack, this attack can never cause a conflict, hence, these attacks can be omitted as well. Tr_2 translates a SETAF into a hypergraph such that for every two arguments a and b there cannot be both a path from a to b and from b to a. Moreover the resulting SETAF does not have any attacks (T, h) with $h \in T$, hence, for every SETAF SF its translation $Tr_2(SF)$ is acyclic.
Note that in general $Tr_2(SF)$ is not redundancy-free (as Figure 3.2 illustrates). However, this is of no concern for our purposes, as with Tr_2 we are primarily interested in conflict-freeness, where redundancies do not play an important role.

Lemma 17. *Let $SF = (A, R)$ be a SETAF, and $SF' = Tr_2(SF)$. Then $cf(SF) = cf(SF')$.*

Proof. It suffices to show that a set $E \subseteq A$ is conflict-free in SF iff it is conflict-free in SF'.
"$\Rightarrow$": Towards contradiction assume otherwise, i.e. $E \in cf(SF)$, but $E \notin cf(SF')$. Since in SF' by construction there are no self-looping arguments, we have $|E| \geq 2$. As E is not conflict-free in SF' there is an attack $(T', h') \in R^{SF'}$ such that $T' \cup \{h'\} \subseteq E$. This means there is an attack $(T, h) \in R$ with $T \cup \{h\} = T' \cup \{h'\}$. But then E is not conflict-free in SF, which is a contradiction.
"$\Leftarrow$": Similarly, assume we have $E \in cf(SF')$, but $E \notin cf(SF)$. If $|E| = 1$, by construction we have that E is conflict-free in SF. If $|E| \geq 2$, as E is not conflict-free in SF there is an attack $(T, h) \in R$ such that $T \cup \{h\} \subseteq E$. This means either there is an attack $(S \setminus \{x\}, x) \in R^{SF'}$ with $S = T \cup \{h\}$ and $x \in S$, or there is some $a \in T \cup \{h\}$ with $a \rightarrowtail^R a$. In the first case E is not conflict-free in SF', which is a contradiction. In the second case we have that $a \notin E$, so $T \cup \{h\} \not\subseteq E$, which is a contradiction as well. $\square$

Lemma 18. *Let $SF = (A, R)$ be a SETAF, and $SF' = Tr_2(SF)$. Then $naive(SF) = naive(SF')$.*

Proof. It suffices to show that a naive extension $E \subseteq A$ of SF is a naive extension in SF', i.e. $E \in naive(SF)$ iff $E \in naive(SF')$.
For the "⇒"-direction assume $E \in naive(SF)$. Note that there are no self-looping arguments in E. By Lemma 17 we have that E is conflict-free in SF'. It remains to show that for every set $S \subseteq A^{SF'}$ with $S \supset E$ we have that $S \notin cf(SF')$. Towards contradiction assume that S is conflict-free in SF', then by Lemma 17 we have that S is conflict-free in SF, but we have $S \supset E$, which is a contradiction to the assumption that E is a naive extension of SF. The same argument applies for the ⇐-direction.

□

These results suffice to establish exactness for the semantics *cf* and *naive* that are based on conflict-free sets. Note that this holds even though in general the translation 'drops' certain arguments. As these arguments are involved in a self-loop, they could not be in any conflict-free set.

Theorem 7. *For $\sigma \in \{cf, naive\}$ we have that Tr_2 is an efficient, exact translation for $\sigma \Rightarrow \sigma$ such that for every SETAF SF its translation $Tr_2(SF)$ is acyclic.*

Proof. Tr_2 for some SETAF SF can be efficiently computed, as for every attack it suffices to check if it is of the form (a, a) and to compute the least argument w.r.t. $\leq$, which can be done in polynomial time.
$Tr_2(SF)$ being an exact translation follows from Lemma 17 and Lemma 18.

□

Corollary 2. *For $\sigma \in \{cf, naive\}$ we have that Tr_2 is an efficient, exact translation for $\sigma \Rightarrow \sigma$ such that such that for every AF F its translation $Tr_2(F)$ is acyclic.*

Proof. Immediate as Tr_2 maps AFs to AFs.

□

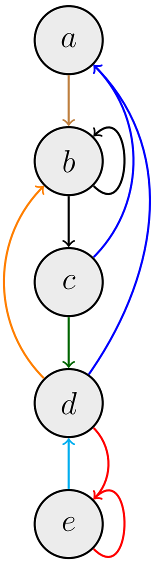
For *stg* semantics, however, even though it is based on conflict-free sets, we have that Tr_2 does not work in general, as the range of the original SETAF does not correspond to the range of the translation. To this end we introduce another translation that uses additional arguments to 'simulate' the original range of the arguments. For an illustration of Tr_3^k see Figure 3.3.

Translation 3. *Let $\leq$ be a total order on the arguments A of a SETAF $SF = (A, R)$. Let*

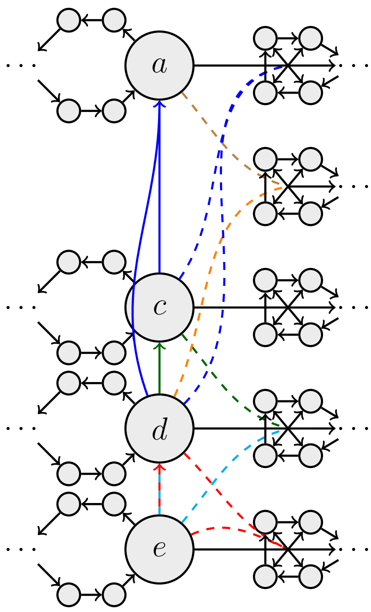
$$\begin{aligned} A^* &= \{a \mid a \in A, (a,a) \notin R\} \\ A'_a &= \{a'_i \mid 1 \leq i \leq 2k+1\}, \text{ for some } a \in A \\ \bar{A}_a &= \{\bar{a}_i \mid 1 \leq i \leq 2k+1\}, \text{ for some } a \in A^*. \end{aligned}$$

The SETAF-translation Tr_3^k for $k \geq 1$ is defined as $Tr_3^k(SF) = (A', R')$, where

$$\begin{aligned} A' &= A^* \cup \bigcup_{a \in A} A'_a \cup \bigcup_{a \in A^*} \bar{A}_a \\ R' &= \{(S \setminus \{x\}, x) \mid (T,h) \in R, S = T \cup \{h\}, x \in S, \forall y \in S \text{ we have } (y,y) \notin R \text{ and} \\ &\quad x \leq y\} \cup \{(a'_i, a'_{i+1}) \mid a \in A, 1 \leq i \leq 2k\} \cup \\ &\quad \{(a'_{2k+1}, a'_1) \mid a \in A\} \cup \{(a, a'_i) \mid a \in A, 1 \leq i \leq 2k+1\} \cup \\ &\quad \{(\bar{a}_i, \bar{a}_{i+1}) \mid a \in A^*, 1 \leq i \leq 2k\} \cup \{(a, \bar{a}_1), (\bar{a}_{2k+1}, a) \mid a \in A^*\} \cup \\ &\quad \{(T, h'_i) \mid (T,h) \in R, 1 \leq i \leq 2k+1, \nexists a \in T \text{ s.t. } (a,a) \in R\} \end{aligned}$$



(a)



(b)

Figure 3.3: Illustration of translation Tr_3^k. We see a SETAF before 3.3a and after 3.3b the translation with $a \leq b \leq c \leq d \leq e$. In 3.3b the labelled arguments $A^* = \{a, c, d, e, f\}$ are in the middle, on their left side there are the arguments $\bar{A}_a$ for each $a \in A^*$ respectively forming an even-cycle, on their right side there are the arguments A'_a for each $a \in A$ respectively forming an odd-cycle (to simulate the original range). Note that in the odd cycles every argument attacks exactly one other argument.

The main construction of Tr_3^k is similar to the construction of Tr_2; the following lemma illustrates that we still have the correspondence between conflict-free sets between a SETAF and its translation.

Lemma 19. *Let $SF = (A, R)$ be a SETAF, and $SF' = Tr_3^k(SF) = (A', R')$. Then for every $E' \subseteq A'$ we have $E' \in cf(SF')$ iff $E = E' \cap A \in cf(SF)$.*

Proof. "⇒": Towards contradiction assume we have $E \in cf(SF')$, but $E \notin cf(SF)$. If $|E| = 1$, by construction we have that E is conflict-free in SF. If $|E| \geq 2$, as E is not conflict-free in SF there is an attack $(T, h) \in R$ such that $T \cup \{h\} \subseteq E$. This means either there is an attack $(S \setminus \{x\}, x) \in R^{SF'}$ with $S = T \cup \{h\}$ and $x \in S$, or there is some $a \in T \cup \{h\}$ with $a \rightarrowtail^R a$. In the first case E is not conflict-free in SF', which is a contradiction. In the second case we have that $a \notin E$, so $T \cup \{h\} \not\subseteq E$, which is a contradiction as well.
"⇐": Similarly, assume otherwise, i.e. $E \in cf(SF)$, but $E \notin cf(SF')$. Since in SF' by construction there are no self-looping arguments, we have $|E| \geq 2$. As E is not conflict-free in SF' there is an attack $(T', h') \in R'$ such that $T' \cup \{h'\} \subseteq E$. This means there is an attack $(T, h) \in R$ with $T \cup \{h\} = T' \cup \{h'\}$. But then E is not conflict-free in SF, which is a contradiction. □

The following lemma illustrates how in Tr_3^k the range of the original arguments is 'simulated' by the odd-cycles formed by the sets A'_a.

Lemma 20. *Let $SF = (A, R)$ be a SETAF and let $E \in cf(SF)$ and let $SF' = Tr_3^k(SF) = (A', R')$. Then for $a \in A$ we have $A'_a \subseteq E^{\oplus}_{R'}$ iff $a \in E^{\oplus}_R$.*

Proof. "⇒": Since the arguments in A'_a form an odd-cycle, they can only be all in the range of E if at least one argument $x \in A'_a$ is attacked by a set $T \subseteq A' \setminus A'_a$. Then by construction we have that T (and hence E) attacks every argument in A'_a. We could either have $T = \{a\}$, then we have $a \in E$ and trivially $a \in E^{\oplus}_R$, or there is an attack $(T, a) \in R$, then since $T \subseteq E$ we also have $a \in E^{\oplus}_R$.
"⇐": Since E is conflict-free, we have that $(a, a) \notin R$. So we either have $a \in E$, then by construction we have $A'_a \subseteq E^{\oplus}_{R'}$, or we have an attack $(T, a) \in R$ such that $T \subseteq E$. Since E is conflict-free in SF also T is conflict-free in SF, so especially there is no $b \in T$ such that $(b, b) \in R$. Hence, there are attacks $(T, x) \in R'$ to every $x \in A'_a$, i.e. $A'_a \subseteq E^{\oplus}_{R'}$. □

Lemma 21. *Let $SF = (A, R)$ be a SETAF and let $SF' = Tr_3^k(SF) = (A', R')$. Then for every extension $E \in stg(SF)$ is there is a corresponding extension $E' \subseteq A'$ with $E' \supseteq E$ such that $E' \in stg(SF')$.*

Proof. Let

$$\begin{aligned} E' &= E \cup \{a'_{2i} \mid a \in A, a \in E^{\oplus}_R, 1 \leq i \leq k\} \cup \\ &\quad \{\bar{a}_{2i} \mid a \in A, (a, a) \notin R, a \in E, 1 \leq i \leq k\} \cup \\ &\quad \{\bar{a}_{2i-1} \mid a \in A, (a, a) \notin R, a \notin E, 1 \leq i \leq k+1\}. \end{aligned}$$

We will now show that $E' \in stg(SF')$. By construction we have $E' \in cf(SF')$. Assume towards contradiction there is a set $S' \subseteq A'$ such that $S' \in cf(SF')$ and $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$. By construction we have that $A^* \cup \bigcup_{a \in A^*} \bar{A}_a \subseteq E'^{\oplus}_{R'}$, so we also have $A^* \cup \bigcup_{a \in A^*} \bar{A}_a \subseteq S'^{\oplus}_{R'}$. Therefore there must be an argument $x \in A'_a$ for some $a \in A$ such that $x \in S'^{\oplus}_{R'}$, but $x \notin E'^{\oplus}_{R'}$. From the latter by Lemma 20 we get that $a \notin E^{\oplus}_{R}$. We have $A'_a \setminus \{x\} \subseteq E'^{\oplus}_{R'}$, since by construction either all arguments of A'_a are in $E'^{\oplus}_{R'}$ or all but one. This means that $A'_a \subseteq S'^{\oplus}_{R'}$. Now let $S = S' \cap A$, then by Lemma 20 we have that $a \in S^{\oplus}_{R}$. Moreover, by Lemma 19 we have $S \in cf(SF)$, and from $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$ by repeatedly applying Lemma 20 we get $S^{\oplus}_{R} \supseteq E^{\oplus}_{R}$, and since $a \in S^{\oplus}_{R}$ but $a \notin E^{\oplus}_{R}$ we get $S^{\oplus}_{R} \supset E^{\oplus}_{R}$, which is a contradiction to the assumption that E is stage in SF. □

Lemma 22. *Let $SF = (A, R)$ be a SETAF and let $SF' = Tr^k_3(SF) = (A', R')$. Then for every extension $E' \in stg(SF')$ we have that $E = E' \cap A \in stg(SF)$.*

Proof. Towards contradiction assume the contrary, i.e. there is an extension $E' \in stg(SF')$ such that $E = E' \cap A \notin stg(SF)$. This means there is a set $S \subseteq A$ with $S \in cf(SF)$ such that $S^{\oplus}_{R} \supset E^{\oplus}_{R}$. Let $a \in S^{\oplus}_{R} \setminus E^{\oplus}_{R}$ and let

$$
\begin{aligned}
S' \;=\; & S \cup \{a'_{2i} \mid a \in A, a \in S^{\oplus}_{R}, 1 \leq i \leq k\} \;\cup \\
& \{\bar{a}_{2i} \mid a \in A, (a, a) \notin R, a \in S, 1 \leq i \leq k\} \;\cup \\
& \{\bar{a}_{2i-1} \mid a \in A, (a, a) \notin R, a \notin S, 1 \leq i \leq k + 1\}.
\end{aligned}
$$

By construction we have $A^* \cup \bigcup_{a \in A^*} \bar{A}_a \subseteq S'^{\oplus}_{R'}$. From $S^{\oplus}_{R} \supset E^{\oplus}_{R}$ and $A^* \cup \bigcup_{a \in A^*} \bar{A}_a \subseteq S'^{\oplus}_{R'}$ by Lemma 20 we get we also get $E'^{\oplus}_{R'} \subseteq S'^{\oplus}_{R'}$. Moreover, since $a \in S^{\oplus}_{R} \setminus E^{\oplus}_{R}$ by Lemma 20 we get $A'_a \subseteq S'^{\oplus}_{R'}$, but $A'_a \not\subseteq S'^{\oplus}_{R'}$, hence $E'^{\oplus}_{R'} \subset S'^{\oplus}_{R'}$, which is a contradiction to the assumption that E' is stage in SF'. □

Theorem 8. *Tr^k_3 is an efficient, acceptance preserving translation for $stg \Rightarrow stg$ such that $mincl(Tr^k_3(SF)) \geq k$ for every SETAF SF and every $k \geq 1$.*

Proof. $Tr^k_3(SF)$ for some SETAF SF can be efficiently computed, as only polynomially many attacks and arguments are added to SF.
Finally, $stg(SF) = \{E \cap A^{SF} \mid E \in stg(Tr^k_3(SF))\}$ follows from Lemma 21 and Lemma 22. □

Corollary 3. *Tr^k_3 is an efficient translation for $stg \Rightarrow stg$ such that $stg(F) = \{E \cap A^F \mid E \in stg(Tr^k_3(F))\}$ and $mincl(Tr^k_3(F)) \geq k$ for every AF F and every $k \geq 1$.*

Proof. Immediate as Tr^k_3 maps AFs to AFs. □

Theorem 9. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the problems $Cred_\sigma$ and $Skept_\sigma$ for an AF F with $mincl(F) \geq k$ have their full complexity, i.e. the complexity results in Table 3.5 hold.*

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	P-c	P-c	Π_2^P-c	Π_2^P-c

Table 3.5: The complexity for SETAFs with no cycles shorter than k arguments with $k \geq 0$ (full complexity).

Proof. The membership for the respective problems follows from the general case. The hardness of $Cred_\sigma$ and $Skept_\sigma$ for $\sigma \in \{grd, stb, com, pref, sem\}$ follows from Corollary 1. The hardness for $Cred_{adm}$ follows from the identity $Cred_{adm} = Cred_{com}$. The hardness for $Cred_{stg}$ and $Skept_{stg}$ follows from Corollary 3. □

As every AF is a SETAF, this result immediately carries over to SETAFs.

Corollary 4. *For $\sigma \in \{cf, adm, naive, stb, com, pref, stg, sem\}$ for SETAFs SF with $mincl(F) \geq k$ the complexity results in Table 3.5 hold.*

Often one is interested in AFs or SETAFs without self-attacks, i.e. without attacks (a, a) or (T, h) with $h \in T$. Self-attacks are cycles of length 1, so we obtain the following corollary by weakening Corollary 4.

Corollary 5. *For $\sigma \in \{cf, adm, naive, stb, com, pref, stg, sem\}$ the problems $Cred_\sigma$ and $Skept_\sigma$ for a SETAF SF without self-attacks have their full complexity, i.e. the complexity results in Table 3.5 hold.*

3.3.2 Maximal Cycle Length

It would be straight forward to define the maximal cycle length of a SETAF analogously to the minimal cycle length as the length of the longest cycle. However, as every SETAF that contains a cycle of length n also contains cycles of length $k * n$ for any $k \geq 1$ that one can get by simply concatenating the cycle k times, this definition is not useful. Instead, we have to rule out these 'repeated' cycles.
In the following other notions of acyclicity we are always interested in cycles where an argument is allowed to appear at most once in the head of an attack that belongs to the cycle, which is why the function *maxcl* will be defined similarly.

Definition 18. *Let SF be a SETAF. Then its* maximal cycle length *$maxcl(SF)$ is defined as the length of the longest cycle $(a_1, a_2, \ldots, a_n, a_1)$ in SF such that $a_i \neq a_j$ for $1 \leq i < j \leq n$, and 0 if SF is acyclic.*

We have a SETAF SF with $maxcl(SF) = 0$ is acyclic, therefore the complexity results for acyclic SETAFs apply. As previously we were interested in the complexity of SETAFs with a minimal cycle length of *at least* k, similarly here we are interested in the complexity of SETAFs with a maximal cycle length of *at most* k.

Lemma 23. *Let $SF = (A, R)$ be a SETAF with $maxcl(SF) \leq 3$ and let $E \in com(SF)$ be a complete extension of SF. Then there is an admissible set $E' \in adm(SF)$ such that $E' \supset E$ iff there is an attack (S, h) with $S \not\subseteq E$ such that $E \cup S$ is admissible.*

Proof. The "$\Leftarrow$"-direction is trivial. For the "$\Rightarrow$"-direction assume $E \in com(SF)$ and $E' \in adm(SF)$ with $E' \supset E$, i.e. there is an argument $a_1 \in E' \setminus E$. As E is complete, we know E does not defend a_1, otherwise we would have $a_1 \in E$. Moreover there is at least one attack (T_1, a_1) towards a_1. From the assumption that E' is admissible we get that E' attacks T_1, i.e. there is an argument $t_1 \in T_1$ such that for an attack (A_2, t_1) we have $A_2 \subseteq E'$. Since $a_1 \notin E$ and E is complete, we know that E does not attack T_1, i.e. there is an argument $a_2 \in A_2$ such that $a_2 \in E' \setminus E$. Now for a_2 the same argument applies as for a_1, i.e. it is attacked by a set T_2 with $t_2 \in T_2$, and it is defended by an attack (A_3, t_2) with $A_3 \subseteq E'$, but $A_3 \not\subseteq E$. This means there is an infinite sequence of sets of arguments $(A_1, A_2, \dots)$, such that A_{i+1} defends a_i with an attack (A_{i+1}, t_i) and $A_i \subseteq E'$, but $a_i \notin E$ for $i \geq 1$. But since we assume the set of attacks R to be finite, we have $A_i = A_j$ for some $i, j \geq 1$. Assume towards contradiction $i > j$, then $(t_{i-1}, a_{i-1}, t_{i-2} \dots, t_j, a_j, t_{i-1})$ is a cycle of length at least 4, which is a contradiction. Symmetrically, if $i < j$, there is a cycle of length at least 4. This means we have $i = j$, i.e. A_i defends itself against the attack (T_i, a_i).
By exhaustively applying this argument we obtain a set S with $S \not\subseteq E$ such that $E \cup S$ is admissible. □

This property of SETAFs with cycles of maximal length of 3 yields an algorithm to verify preferred extensions in polynomial time, which in turn means skeptical acceptance is in coNP.

Lemma 24. *Ver_{pref} for SETAFs $SF = (A, R)$ with $maxcl(SF) \leq 3$ is in P.*

Proof. Given a set $S \subseteq A$ one can check in polynomial time whether S is complete. Moreover, we can check for each attack $(T, h) \in R$ whether $T \not\subseteq S$ and $T \cup S$ is admissible, which can clearly be done in polynomial time, as there are only polynomially many attacks and Ver_{adm} is in P for SETAFs. By Lemma 23 we know that this check suffices to decide whether S is preferred or not. □

Lemma 25. *$Skept_{pref}$ for SETAFs $SF = (A, R)$ with $maxcl(SF) \leq 3$ is coNP-complete.*

Proof. The membership follows from the fact that if the problem Ver_σ is in a complexity class $\mathcal{C}$, then $Skept_\sigma$ is in $\mathsf{coNP}^{\mathcal{C}}$. This is because the complementary problem, i.e. deciding whether an argument $a \in A$ is not skeptically accepted, can be decided by guessing a set $S \subseteq A$ with $a \notin A$ and using a $\mathcal{C}$ oracle to verify that S is a σ extension. By Lemma 24 we have that Ver_{pref} is in P.
The hardness follows from the fact that the standard reduction for AFs already proves coNP-hardness for $Skept_{pref}$, and all its cycles have length at most 2. □

In the following we will stay with *pref* semantics and establish that for SETAFs SF with $maxcl(SF) \leq 4$ the problem $Skept_{pref}$ is already Π_2^P-hard. It is not know whether this also holds for AFs. To this end we introduce a new reduction from the Π_2^P-complete $QBF_\forall^2$ problem with at least 2 clauses such that every clause has positive and negative literals to $Skept_{pref}$ such that for every $QBF_\forall^2$-formula the respective SETAF that is the result of the reduction has no cycles that consist of more than 4 arguments.
The main difference between these reductions and the standard reductions is that here a clause is jointly attacked by the *duals* of the literals in this clause. This way, only if the corresponding interpretation makes all duals of a clause's literals true, the clause is attacked. If at least one clause is attacked, $\bar{\varphi}$ cannot be attacked and φ cannot be defended.
The attacks from the atoms and their negations ensure that in every admissible set that contains φ at least one of x and $\bar{x}$ is in the set (otherwise φ would not be defended against that attack).
SF_3^φ shows the 'basic' variation of this reduction, i.e. a construction that shows NP-hardness for $Cred_\sigma$ for $\sigma \in \{adm, com, pref, stb\}$ and the coNP-hardness for $Skept_{stb}$ (not explicitly shown here) and SF_4^Φ shows the Π_2^P-hardness for $Skept_{pref}$. Note that the longest cycle in SF_4^Φ is of length 4.
For illustrations see Figure 3.4 and Figure 3.5 respectively.

Reduction 3. *Let φ be a CNF-formula consisting of a set of clauses C over a set of propositional atoms X. We define the SETAF $SF_3^\varphi = (A, R)$, where*

$$\begin{aligned} A &= \{\varphi, \bar{\varphi}\} \cup C \cup X \cup \bar{X}, \\ R &= \{(x, \bar{x}), (\bar{x}, x) \mid x \in X\} \cup \{(\{x \mid \bar{x} \in c\} \cup \{\bar{x} \mid x \in c\}, c) \mid c \in C\} \ \cup \\ &\quad\ \{(\{c \mid c \in C\}, \bar{\varphi}), (\bar{\varphi}, \varphi)\} \cup \{(\{x, \bar{x}\}, \varphi) \mid x \in X\}. \end{aligned}$$

Lemma 26. *φ is in an admissible set of SF_3^φ iff φ is satisfiable.*

Proof. "$\Rightarrow$": Assume φ is in an admissible set S, in order to defend it we have that for every $x \in X$ at least one of x or $\bar{x}$ is attacked. At the same time, as S is conflict-free, and as x and $\bar{x}$ only attack the respective other, exactly one of x or $\bar{x}$ in in S, i.e. $X \cap S$ corresponds to an interpretation over the atoms X. Likewise $\bar{\varphi}$ has to be attacked, i.e. we have $C \subseteq S$, which means that for every attack towards an argument $c \in C$ we have that at least one argument attacking c is not in S. As established, this means its dual is in S, hence, for every clause at least one of its literals is true in the corresponding interpretation, i.e. φ is satisfiable.
"$\Leftarrow$": Assume φ is satisfiable, let $\mathcal{I}$ be a model of φ and let $S = \{x \mid x \in X, \mathcal{I}(x) = 1\} \cup \{\bar{x} \mid x \in X, \mathcal{I}(x) = 0\} \cup C \cup \{\varphi\}$. Then S is admissible, as every argument $x \in X \cap S$ and $\bar{x} \in \bar{X} \cap S$ defends itself, for every argument $c \in C$ there is at least one argument that attacks the attack towards c (as φ is satisfiable, $\mathcal{I}$ cannot make all of the duals of the literals of a clause true), so φ is defended. □

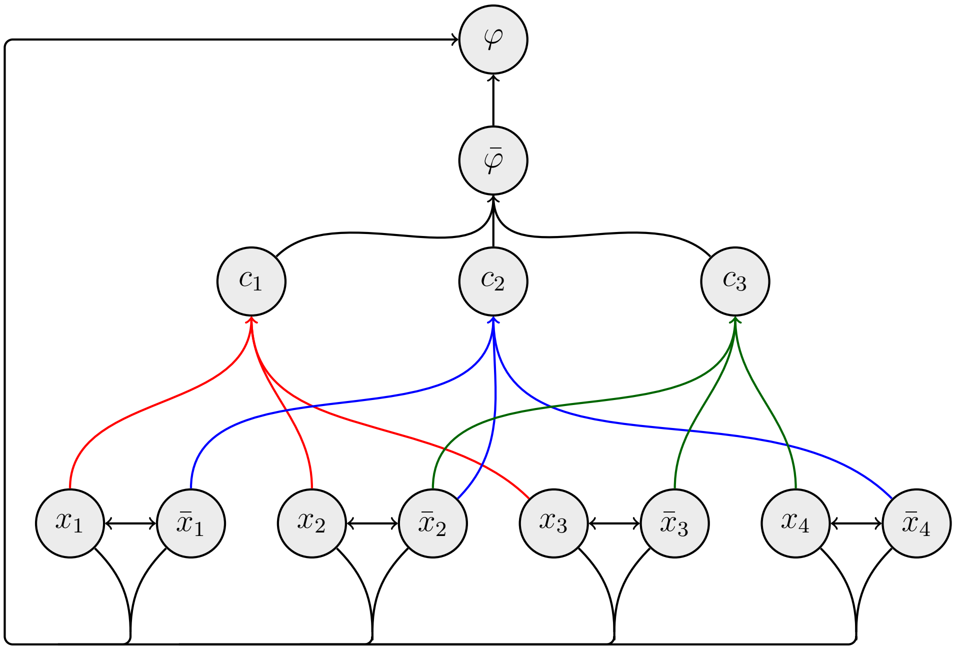


Figure 3.4: Illustration of SF_3^{φ} for a CNF-formula φ with $X = \{x_1, x_2, x_3, x_4\}$, and $\varphi = \{\{\bar{x}_1, \bar{x}_2, \bar{x}_3\}, \{x_1, x_2, x_4)\}, \{x_2, x_3, \bar{x}_4\}\}$. Note that the attacks from the $\{y, \bar{y}\}$ and $\{z, \bar{z}\}$ towards φ have exactly two arguments in the tail each, they overlap in this illustration only in the interest of presentability.

Reduction 4. *Let $\Phi = \forall Y \exists Z C$ be a $QBF_{\forall}^2$-formula consisting of a set of clauses C over sets of propositional atoms Y and Z. We define the SETAF $SF_4^{\Phi} = (A, R)$, where*

$$\begin{aligned} A &= \{\varphi, \bar{\varphi}\} \cup C \cup Y \cup \bar{Y} \cup Z \cup \bar{Z}, \\ R &= \{(x, \bar{x}), (\bar{x}, x) \mid x \in Y \cup Z\} \cup \{(\{x \mid \bar{x} \in c\} \cup \{\bar{x} \mid x \in c\}, c) \mid c \in C\} \cup \\ &\quad \{(\{c \mid c \in C\}, \bar{\varphi}), (\bar{\varphi}, \varphi)\} \cup \{(\{x, \bar{x}\}, \varphi) \mid x \in X\} \cup \\ &\quad \{(\bar{\varphi}, z), (\bar{\varphi}, \bar{z}) \mid z \in Z\}. \end{aligned}$$

Lemma 27. *φ is in every preferred extension of SF_4^{Φ} iff Φ is true.*

Proof. "$\Rightarrow$": Every partial interpretation $\mathcal{I}_Y$ corresponds to an admissible set in SF_4^{Φ}, whereas arguments $z \in Z$ can only be in an admissible set if $\bar{\varphi}$ is attacked. In order to attack $\bar{\varphi}$, as established in Lemma 26, we need for every $z \in Z$ exactly one of z and $\bar{z}$ in an admissible set, and moreover any admissible set attacking $\bar{\varphi}$ has to correspond to a satisfying assignment. Note that every clause c has at least one literal from $Z \cup \bar{Z}$ (otherwise the problem would be trivially false), therefore we cannot have $\bar{\varphi}$ in an admissible set. As every partial interpretation $\mathcal{I}_Y$ corresponds to an admissible set and

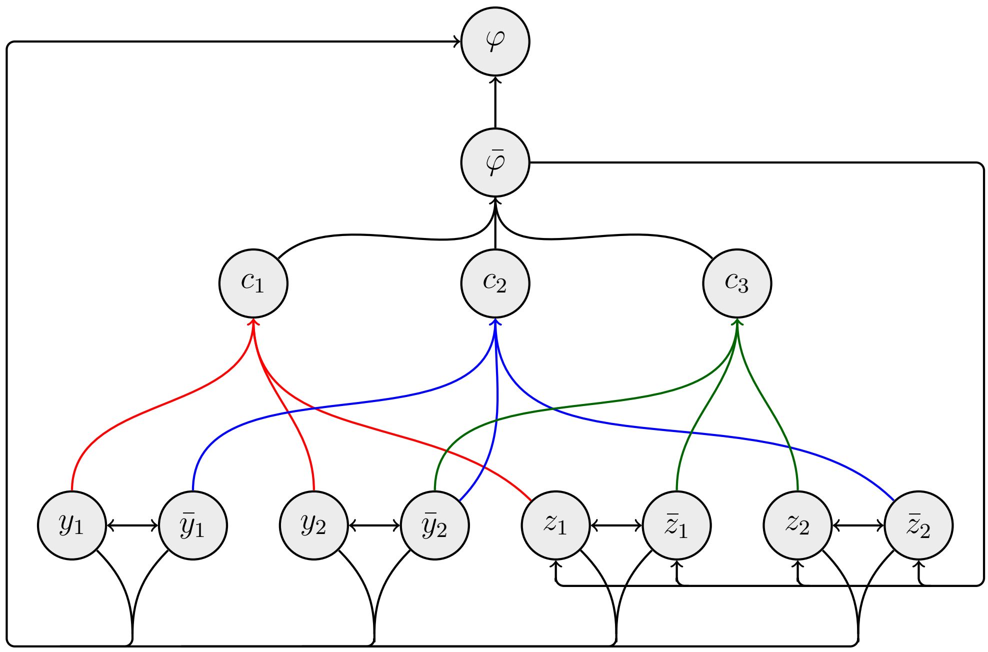


Figure 3.5: Illustration of SF_4^Φ for $\Phi = \forall Y \exists Z \varphi(Y, Z)$ with $Y = \{y_1, y_2\}$, $Z = \{z_1, z_2\}$, and $\varphi = \{\{\bar{y}_1, \bar{y}_2, \bar{z}_1\}, \{y_1, y_2, z_2)\}, \{y_2, z_1, \bar{z}_2\}\}$. Note that the attacks from the $\{y, \bar{y}\}$ and $\{z, \bar{z}\}$ towards φ have exactly two arguments in the tail each, they overlap in this illustration only in the interest of presentability.

φ is only in admissible sets that correspond to satisfying assignments, Φ is true.
"$\Leftarrow$": As Φ is true, we know that for every partial assignment I_Y there is a partial assignment I_Z such that $I_Y \cup I_Z$ make φ true. This means for each such assignment there is a subset-maximal admissible set such that no $c \in C$ is attacked, and as φ is then defended against all attacks towards it, we have that φ is in all these preferred extensions. Moreover we have that all preferred extensions are of this form, as for an argument $x \in Z \cup \bar{Z}$ to be in an admissible set the argument $\bar{\varphi}$ has to be attacked, which in term means every argument $c \in C$ is in every preferred extension. □

Theorem 10. *For the problems $Cred_\sigma$ and $Skept_\sigma$ for AFs and SETAFs SF with $maxcl(SF) \leq k$ the complexity results in Table 3.6 hold.*

Proof. First note that results for $k = x$ are a lower bound for the results for $k \geq x + 1$, as every (SET)AF SF with $maxcl(SF) \leq x$ also has $maxcl(SF) \leq x + 1$. For $k = 0$ the problem coincides with reasoning in acyclic AFs (see Theorem 1).
As every AF or SETAF SF with $maxcl(SF) \leq 1$ is even-cycle-free, the complexity of even-cycle-free (SET)AFs (see Table 3.3) is an upper bound. Finally, the fact that $Cred_{stg}$ and $Skept_{stg}$ are Σ_2^P-hard and Π_2^P-hard respectively even for AFs with only a single cycle of length 1 (see 2) concludes the proof for $k \leq 1$.

k		cf	adm	$naive$	stb	$pref$	com	grd	stg	sem
0	$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	P-c	P-c
	$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	P-c	P-c
1	$Cred_\sigma$	in L	P-c	in L	P-c	P-c	P-c	P-c	Σ_2^P-c	P-c
	$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	Π_2^P-c	P-c
2, 3	$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	trivial	in L	coNP-c	coNP-c	P-c	P-c	Π_2^P-c	Π_2^P-c
≥ 4	$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c [2]	P-c	P-c	Π_2^P-c	Π_2^P-c

Table 3.6: The complexity AFs and SETAFs SF with $maxcl(SF) \leq k$, i.e. the complexity if it is guaranteed that the longest cycle with no repeating arguments is of length at most k.

The membership of all problems but $Skept_{pref}$ for $2 \leq k \leq 3$ immediately follows from the general case. The hardness of $Cred_\sigma$ for $\sigma \in \{adm, pref, stb, com, sem\}$ and $Skept_{sem}$ with $2 \leq k \leq 3$ follows from the fact that the respective standard reductions have a maximal cycle length of 2. The membership and hardness for $Skept_{pref}$ with $2 \leq k \leq 3$ follow from Lemma 25. The hardness for $Skept_{pref}$ for $k \geq 4$ for SETAFs follows from Lemma 27. Finally, note that the hardness for $Skept_{pref}$ for $k \geq 5$ for AFs follows from the fact that the respective standard reduction has a maximal cycle length of 5 (e.g. a cycle $(z, \bar{z}, c, \varphi, \bar{\varphi}, z)$ with $z \in Z$, $c \in C$, and $\bar{z} \in c$). □

Just like with acyclicity, it is easy to check if a given directed graph has a cycle length of at most 1. For this it suffices to iteratively remove leafs or nodes that only have themselves as a successor, iff every node gets removed we know the graph has no cycle that consists of more than 1 nodes. As it suffices to check the primal graph, this carries over to SETAFs. This means we identified another tractable fragment for both AFs and SETAFs.

Theorem 11. *A maximal cycle length of* 1 *is a tractable fragment for SETAFs for* $\sigma \in \{cf, adm, naive, stb, pref, com, grd, sem\}$.

Note that this subsumes the tractable fragment of acyclic SETAFs for the given semantics, and it also applies for AFs.

3.4 Other Notions of Acyclicity

3.4.1 Definitions and Basic Properties

In the literature there are various notions of acyclicity for hypergraphs to be found. In the following we adapt these notions for SETAFs, for illustrations of the respective "degrees" of acyclicity see Figure 3.6.

[2] It is unknown if the problem $Skept_{pref}$ is in coNP or if it is Π_2^P-hard for AFs F with $maxcl(F) \leq 4$; we know it is Π_2^P-complete for SETAFs SF with $maxcl(SF) \leq 4$.

Definition 19 (cf. [Ber73, p. 391])**.** *A cycle $P = (a_1, (S_1, a_2), \dots, (S_n, a_1), a_1)$ is a* Berge-cycle *if*

(a) all $a_1, \dots, a_n$ are distinct,

(b) $n \geq 2$, i.e. there are at least two edges involved.

A SETAF is called Berge-cyclic*, if it contains a Berge-cycle, otherwise it is called* Berge-acyclic*.*

From this definition (and Lemma 28) it follows that a Berge-acyclic SETAF SF is acyclic iff it has no self-attacks. Equivalently, this means that SF is Berge-acyclic iff it has $maxcl(SF) \leq 1$. Since we have already investigated the complexity of SETAFs with this property in Section 3.3.2, the results for Berge-acyclic SETAFs immediately follow.

Definition 20 (cf. [Fag83, p. 521])**.** *A cycle $P = (a_1, (S_1, a_2), \dots, (S_n, a_1), a_1)$ is a* weak β-cycle *if*

(a) all $a_1, \dots, a_n$ are distinct,

(b) $S_i \cap S_j = \emptyset$ for all $1 \leq i, j \leq n$ with $i \neq j$, and

(c) $n \geq 3$, i.e. there are at least three edges involved.

A SETAF is called β-cyclic*, if it contains a weak β-cycle, otherwise it is called* β-acyclic*.*

Definition 21 (cf. [Fag83, p. 524])**.** *A cycle $P = (a_1, (S_1, a_2), \dots, (S_n, a_1), a_1)$ is a* γ-cycle *if*

(a) all $a_1, \dots, a_n$ are distinct,

(b) $S_i \cap S_j = \emptyset$ for all $1 \leq i, j < n$ with $i \neq j$, and

(c) $n \geq 3$, i.e. there are at least three edges involved.

A SETAF is called γ-cyclic*, if it contains a γ-cycle, otherwise it is called* γ-acyclic*.*

Note, that the only difference between a weak β-cycle and a γ-cycle is that condition (b) holds "for all $1 \leq i, j < n$" instead of "for all $1 \leq i, j \leq n$".

Definition 22. *A cycle $P = (a_1, (S_1, a_2), \dots, (S_n, a_1), a_1)$ is a* γ'-cycle *if*

(a) all $a_1, \dots, a_n$ are distinct,

(b) $S_i \cap S_j = \emptyset$ for all $1 \leq i, j < n$ with $i \neq j$, and

(c) $n \geq 2$, i.e. there are at least two edges involved.

A SETAF is called γ'-cyclic*, if it contains a γ'-cycle, otherwise it is called* γ'-acyclic.

The only difference between a γ-cycle and a γ'-cycle is that condition (c) says "$n \geq 2$" instead of "$n \geq 3$".

Lemma 28. *If a SETAF $SF = (A, R)$ contains a cycle P of length n, then it contains a cycle $P' = (a_1, (S_1, a_2), \ldots, (S_m, a_m), a_1)$, with*

(a) all $a_1, \ldots, a_m$ are distinct, and

(b) $m \leq n$.

Proof. Let $P = (a_1, (S_1, a_2), \ldots, a_k, (S_k, a_{k+1}), \ldots, a_l, (S_l, a_{l+1}), \ldots, (S_n, a_1), a_1)$ be a cycle with $a_k = a_l$. Then we can construct a shorter cycle $P' = (a_k, (S_k, a_{k+1}), \ldots, (S_{l-1}, a_l), a_l)$. By iterating this construction we end up with a cycle where each argument appears at most once in the head of an attack. □

Lemma 29. *We have that (classical) acyclicity $\Rightarrow$ Berge-acyclicity $\Rightarrow$ γ'-acyclicity $\Rightarrow$ γ-acyclicity $\Rightarrow$ β-acyclicity.*

Proof. This follows directly from the respective definitions. □

Figure 3.6 shows that the converse does not hold in all cases.

One of the the main differences between Berge-cycles, weak β-cycles, γ-cycles, and γ'-cycles on the one hand, and classical cycles on the other hand, is that the former require the cycle to have a minimum length of 2 or 3 respectively. This restriction on its own does not affect the complexity, as we have shown in Section 3.3.1.
The standard reductions, that use no cycles of bigger length, still prove hardness for the respective decision problems, as for AFs the notions of Berge-acyclicity, β-acyclicity, γ-acyclicity coincides with acyclicity. For the remaining problems we will provide separate proofs for their complexity.
Another difference is that for our new notions of acyclicity we require the involved attacks to be (partly) pairwise disjoint. The effect of this restrictions will be illustrated by the examinations of the complexity of the respective decision problems for our new notions of acyclicity. In the following we will give an alternative approach to generalize acyclicity, that is to view a cycle as a series of attacks.

Definition 23. *Let $SF = (A, R)$ be a SETAF. An* attack path *P of length n is a sequence of sets of arguments and sets of attacks*

$$P = (A_1, R_1, A_2, R_2, \ldots, R_n, A_{n+1}),$$

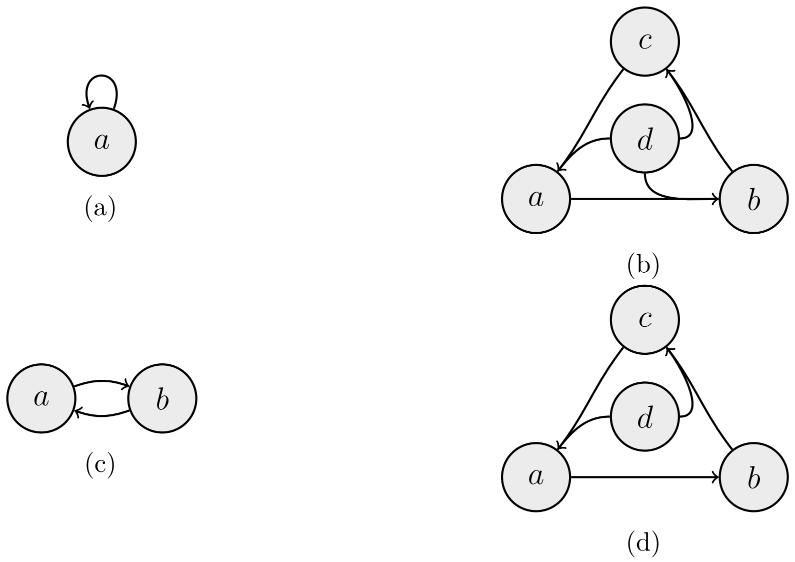


Figure 3.6: Examples for (a) a (classically) cyclic, but Berge-acyclic SETAF, (b) a Berge-cyclic, but γ'-acyclic SETAF, (c) a γ'-acyclic, but γ-cyclic SETAF, and (d) a γ-cyclic, but β-acyclic SETAF.

where $R_i = \{(S_{i,1}, a_{i,1}), \ldots, (S_{i,m_i}, a_{i,m_i})\}$*,* $A_i \subseteq A$*,* $R_i \subseteq R$*, and for* $1 \leq i \leq n$ *it holds that* S_i *attacks* ***some*** *argument in* A_{i+1} *and all attacks in* R_i *start from arguments in* A_i*, i.e.* $\bigcup_{(S_{i,j},a_{i,j})\in R_i} S_{i,j} \subseteq A_i$ *and* $A_{i+1} \cap \bigcup_{(S_{i,j},a_{i,j})\in R_i} \{a_{i,j}\} \neq \emptyset$*.*
An attack path P *of length* n *is an* attack cycle *if* $A_1 = A_{n+1}$*.*
SF is called attack-cyclic *if it contains an attack cycle, otherwise it is called* attack-acyclic *or* attack-cycle-free*.*

One can easily see that every attack-cyclic SETAF is cyclic and vice versa. For an illustration of the next definition see Figure 3.7.

Definition 24. *Let* $SF = (A, R)$ *be a SETAF. A* strong attack path P *of length* n *is a sequence of sets of arguments and sets of attacks*

$$P = (A_1, R_1, A_2, R_2, \ldots, R_n, A_{n+1}),$$

where $R_i = \{(S_{i,1}, a_{i,1}), \ldots, (S_{i,m_i}, a_{i,m_i})\}$*,* $A_i \subseteq A$*,* $R_i \subseteq R$*, and for* $1 \leq i \leq n$ *it holds that* S_i *attacks* ***all*** *arguments in* A_{i+1} *and all attacks in* R_i *start from arguments in* A_i*, i.e.* $\bigcup_{(S_{i,j},a_{i,j})\in R_i} S_{i,j} \subseteq A_i$ *and* $A_{i+1} = \bigcup_{(S_{i,j},a_{i,j})\in R_i} \{a_{i,j}\}$*.*
A strong attack path P *of length* n *is a* strong attack cycle *if* $A_1 = A_{n+1}$*.*
SF is called strong-attack-cyclic *if it contains a strong attack cycle, otherwise it is called* strong-attack-acyclic *or* strong-attack-cycle-free*.*

Both attack-acyclicity and strong-attack-cyclicity coincide with classical acyclicity when restricted to AFs.

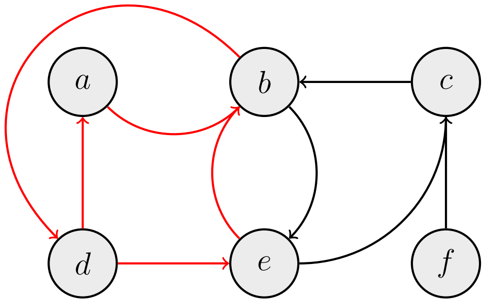


Figure 3.7: Example of a strong attack-cyclic SETAF; the (only) strong attack cycle of this SETAF $(\{a,e\},\{(\{a,e\},b)\},\{b\},\{(b,d)\},\{d\},\{(d,a),(d,e)\},\{a,e\})$ is highlighted in red color.

3.4.2 Generalizations of Acyclicity that are Unsuitable for SETAFs

One might also try to adapt the notion of α-acyclicity for SETAFs. However, it will turn out that this approach does not end up with a notion that gives us acyclicity when applied to AFs. In the following we adapt α-acyclicity as given in [Fag83, p. 517-519] for SETAFs. Note that for directed graphs, connectivity can mean three different properties, which is why we end up with three different notions of α-acyclicity (none of which have the desired property of capturing acyclicity in the case of AFs).

Definition 25. *A sequence of edges $P = ((S_1, a_1), \ldots, (S_n, a_n))$ in a SETAF is a* weak path *of length n, if $(S_i \cup \{a_i\}) \cap (S_{i+1} \cup \{a_{i+1}\}) \neq \emptyset$ for $1 \leq i \leq n-1$. Two arguments x and y are* weakly connected *if there is a weak path $P = ((S_1, a_1), \ldots, (S_n, a_n))$ with $x \in (S_1 \cup \{a_1\})$ and $y \in (S_n \cup \{a_n\})$.*
Two arguments x and y are connected *if there is a path from x to y or from y to x.*
Two arguments x and y are strongly connected *if there is a path from x to y and from y to x.*

Definition 26. *A set of edges of a SETAF are a* (weakly, strongly) connected set *if every pair of arguments occurring in the edges is (weakly, strongly) connected.*

Definition 27 (cf. [Pol17, p. 135])**.** *The* minimal form *of a SETAF SF is obtained by removing all redundant edges from SF.*

Definition 28. *Let $SF = (A, R)$ be a SETAF and let $M \subseteq A$ be a set of arguments. Then we call the minimal form of $\{(S \cap M, a) \mid (S, a) \in R, S \cap M \neq \emptyset, a \in M\}$ the* set of partial edges generated by *M on SF.*

Definition 29. *Let R' be a (weakly, strongly) connected set of partial edges and let A' be the set of all arguments occurring in R'. Let $(S_1, a_1), (S_2, a_2) \in R'$ and let $Q =$*

$(S_1 \cup \{a_1\}) \cap (S_2 \cup \{a_2\})$. *Then we call* Q *a* (weak, strong) articulation set*, if the set of arguments occurring in the set of partial edges generated by* $A' \setminus Q$ *on* (A', R') *is not (weakly, strongly) connected.*

Definition 30. *A* (weak, strong) block *of a SETAF SF is the minimal form of a (weakly, strongly) connected set of partial edges with no (weak, strong) articulation set. A block is called* trivial *if it has at most one edge.*

Definition 31. *A SETAF SF is called* (weakly, strongly) α-acyclic*, if all its (weak, strong) blocks are trivial, otherwise it is called* (weakly, strongly) α-cyclic*.*

Lemma 30. *We have that weak α-acyclicity $\Rightarrow$ α-acyclicity $\Rightarrow$ strong α-acyclicity.*

Proof. We show that every non-trivial strong block is a non-trivial block, and every non-trivial block is a non-trivial weak block.
First assume towards contradiction some non-trivial strong block $B = (A', R')$ was no non-trivial block. B is strongly connected and thus connected, this means that B has an articulation set. But this articulation set is also a strong articulation set, which is a contradiction to the assumption that B was a strong block.
Likewise assume some non-trivial block $B = (A', R')$ was no non-trivial weak block. B is connected and thus weakly connected, this means that B has a weak articulation set. But this weak articulation set is also an articulation set, which is a contradiction to the assumption that B was a block. □

The following examples show undesired properties for the different notions of α-acyclicity, i.e. that none of the notions are a generalization of classic acyclicity in the case of an AF. Furthermore they illustrate that the implications from Lemma 30 are proper.

Example 2. *The AF in Figure 3.8a is strong α-acyclic, but not α-acyclic or acyclic. It is strong α-acyclic, since the only candidate for a non-trivial strong block is the set of partial edges generated by $\{A, B, C\}$ (i.e. the AF itself), but it has three articulation sets: $\{A\}$, $\{B\}$, and $\{C\}$. It is not α-acyclic or weak α-acyclic, because the same set (the AF itself) is a non-trivial block and a non-trivial weak block. The fact that it is strong α-acyclic but clearly not (classically) acyclic leads us the the conclusion that strong α-acyclicity is not a generalization of (classic) acyclicity.*
The AF in Figure 3.8b is strong α-acyclic, α-cyclic, but not cyclic. It is strong α-acyclic, since there are no strongly connected node generated sets of partial edges with at least two edges, and therefore no candidates for non-trivial strong blocks. It is α-cyclic, since it has a non-trivial block, which is the set of partial edges generated by $\{a, b, c\}$ (i.e. the AF itself). It is connected and has no articulation set, as removing every articulation set-candidate (that is $\{a\}$, $\{b\}$, $\{c\}$, $\{a, b\}$, $\{b, c\}$, and $\{a, c\}$) does not destroy the connectivity. The fact that it is α-cyclic, but clearly not (classically) cyclic leads us to the conclusion that α-cyclicity is not a generalization of (classic) acyclicity as well.
The AF in Figure 3.8c is α-acyclic, but not weak α-acyclic or cyclic. It is α-acyclic, since there are no connected node generated sets of partial edges with at least two edges,

and therefore no candidates for non-trivial blocks. It is weak α-cyclic, since it has a non-trivial weak block, which is the set of partial edges generated by $\{a, b, c, d\}$ (i.e. the AF itself). It is weakly connected and has no weak articulation set, as removing every weak articulation set-candidate does not destroy the weak connectivity. The fact that it is weak α-cyclic, but clearly not (classically) cyclic finally leads us to the conclusion that also weak α-cyclicity is not a generalization of (classic) acyclicity.

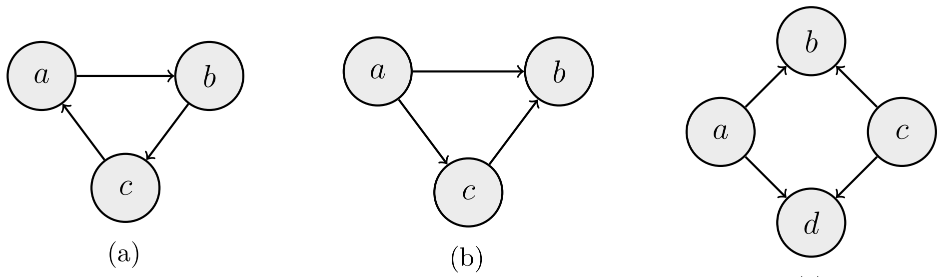


Figure 3.8: Various AFs illustrating undesired properties for α-acyclicity

For our definitions of β-, γ- and γ'-acyclicity we introduced the additional constraint for cycles that the tails of the involved attacks have to be disjoint. The 'original' definitions for β- and γ-acyclicity on general (undirected) hypergraphs (e.g. as described in [Fag83]) however have a different restriction, that is in the intersection of any two adjacent edges there must be a node that does not appear in any other edge than these two. This proposes a different definition for acyclicity in SETAFs, which, as we will show, coincides with Berge-acyclicity and does therefore not add any additional expressiveness. For the purpose of continuity the respective equivalent for SETAFs will be called δ-acyclicity.

Definition 32. *A cycle $P = (a_1, (S_1, a_2), \ldots, (S_n, a_1), a_1)$ is a* δ-cycle *if*

(a) all $a_1, \ldots, a_n$ are distinct,

(b) $a_i \notin S_j$ for all $1 \leq i, j < n$ with $i \neq j$,

(c) $n \geq 2$, i.e. there are at least two edges involved.

A SETAF is called δ-cyclic, *if it contains a δ-cycle, otherwise it is called* δ-acyclic.

From this and the preceding definitions we immediately get the following lemma.

Lemma 31. *Every γ'-cycle is a δ-cycle and every δ-cycle is a Berge-cycle. Moreover every Berge-cycle of length 2 is a δ-cycle.*

This puts the new definition into perspective and makes it clear why we chose the name δ-cycle, as we have that (classical) acyclicity $\Rightarrow$ Berge-acyclicity $\Rightarrow$ δ-acyclicity $\Rightarrow$

γ'-acyclicity $\Rightarrow$ γ-acyclicity $\Rightarrow$ β-acyclicity. However, the following lemma shows that the notion of δ-acyclicity does not add any meaningful information to our examinations, as even Berge-acyclicity $\Leftrightarrow$ δ-acyclicity holds.

Lemma 32. *Every δ-acyclic SETAF is Berge-acyclic.*

Proof. We show that the existence of a Berge-cycle in a SETAF implies the existence of a δ-cycle. Assume there is a Berge-cycle $P = (a_1, (S_1, a_2), \ldots, (S_n, a_1), a_1)$ that is not a δ-cycle. From Lemma 31 we get that the length of P is at least 3. Since P is no δ-cycle, there is some a_k with $1 \leq k < n$ such that $a_k \in S_j$, where $j \neq k$ and $j \neq n$. If $k < j$, then $P' = (a_1, (S_1, a_2), \ldots, a_k, (S_j, a_{j+1}), a_{j+1}, \ldots, (S_n, a_1), a_1)$ is a Berge-cycle with $|P'| < |P|$. If $k > j$, then $P'' = (a_k, (S_j, a_{j+1}), \ldots, (S_{k-1}, a_k), a_k)$ is a Berge-cycle with $|P''| < |P|$. Note that in both cases we have that $|P'| \geq 2$ and $|P''| \geq 2$. By Lemma 31 we get that any Berge-cycle of length 2 is a δ-cycle. By repeatedly applying this construction, we get that if there is a Berge-cycle P of length $|P| \geq 3$ in a SETAF, then either it is a δ-cycle, or there is a δ-cycle of shorter length in this SETAF. □

3.4.3 Complexity

As already established, a SETAF SF is Berge-acyclic iff $maxcl(SF) \leq 1$, therefore by Theorem 10 the following result holds.

Theorem 12. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the complexity of the problems $Cred_\sigma$ and $Skept_\sigma$ for Berge-acyclic SETAFs coincides with the complexity of even-cycle-free SETAFs, i.e. the complexity results in Table 3.3 hold.*

To prove the hardness for different kinds of acyclicity we introduce the following translation for SETAFs. An illustration for this translation can be found in Figure 3.9. The construction from Translation 4 adds one argument to the SETAF that "joins" all attacks.

Translation 4. *Let $SF = (A, R)$ be a SETAF. The SETAF translation Tr_4 is defined as $Tr_4(SF) = (A', R')$ with*

$$\begin{aligned} A' &= A \cup \{a^*\}, \\ R' &= \{(T \cup \{a^*\}, h) \mid (T, h) \in R\} \end{aligned}$$

Intuitively, Tr_4 "eliminates" all cycle-variants that require any two of its attacks' tails to be disjoint.

Lemma 33. *For every SETAF SF we have $Tr_4(SF)$ is strong-attack-cycle-free. Moreover, if we have $mincl(SF) \geq 3$, then SF is also γ'-acyclic.*

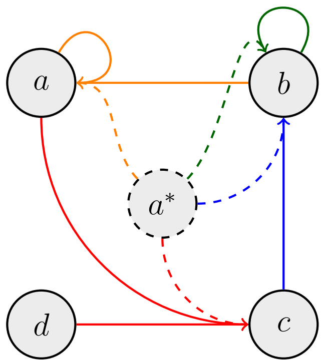


Figure 3.9: SETAF illustrating the construction from Definition 4. Dashed lines indicate the added node and the added parts of the attacks.

Proof. For any attack (T, h) to be part of a strong-attack-cycle, every $a \in T$ has to be attacked by other arguments. Since a^* is not attacked, but part of every tail T, no attack can be part of a strong-attack-cycle.
If $mincl(SF) \geq 3$, then every cycle $P = (a_1, (S_1, a_2), \dots, (S_n, a_1), a_1)$ in SF has $S_1 \cup S_2 \cup S_3 \supseteq \{a^*\}$, and is therefore no γ'-cycle. □

To show that the extensions of a SETAF SF and $Tr_4(SF)$ correspond, we first prove that every "old" extension, that is every extension E of SF, has a corresponding "new" extension in $Tr_4(SF)$ that results from adding a^* to E. For semantics where the newly added argument does not have to be in every extension (such as conflict-free, admissible, naive, and stage), there can be more extensions in the general case (that do not necessarily contain the newly added argument).

Lemma 34. *Let $\sigma \in \{cf, adm, naive, stb, pref, com, grd, stg, sem\}$ be a semantics and let $SF = (A, R)$ be a SETAF and let $SF' = Tr_4(SF) = (A', R')$. Then for any set $S \subseteq A$ it holds that $S \in \sigma(SF)$ iff $S' = (S \cup \{a^*\}) \in \sigma(SF')$.*

Proof. First note that a^* is not attacked in SF'. We will now consider the rest of the proof for each semantics individually.

(a) For $\sigma = cf$:
Let $(T, h) \in R$ be an arbitrary attack. Since $S \in cf(SF)$ we know that some $b \in T \cup \{h\}$ is not in S, therefore it is not in S', hence $S' \in cf(SF')$. The same argument applies for the other direction.

(b) For $\sigma = adm$:
We have established that $S \in cf(SF)$ iff $S' \in cf(SF')$. Note that S and S' attack the same arguments, i.e. $S^{\oplus}_R = S'^{\oplus}_{R'} \setminus \{a^*\}$ (and, hence, defend the same arguments). Furthermore a^* is not attacked in SF' and therefore defended by every set.

(c) For $\sigma = naive$:
We have established that $S \in cf(SF)$ iff $S' \in cf(SF')$. Assume $S \in naive(SF)$ and there is an argument $b \notin S'$ such that $S' \cup \{b\} \in cf(SF')$. Then $S \cup \{b\} \in cf(SF)$, which contradicts the assumption that S is a naive extension for SF. The same argument applies for the other direction.

(d) For $\sigma = stb$:
We have that $S \in cf(SF)$ iff $S' \in cf(SF')$ and $S^{\oplus}_R = S'^{\oplus}_{R'} \setminus \{a^*\}$.

(e) For $\sigma = pref$:
We have that $S \in adm(SF)$ iff $S' \in adm(SF')$. Assume $S \in pref(SF)$ and there is an argument $b \notin S'$ such that $S' \cup \{b\} \in adm(SF')$. Then $S \cup \{b\} \in adm(SF)$, which contradicts the assumption that S is a preferred extension for SF. The same argument applies for the other direction.

(f) For $\sigma = com$:
We have that $S \in adm(SF)$ iff $S' \in adm(SF')$. Assume $S \in com(SF)$ and there is an argument $b \notin S'$ such that b is defended by S'. Then b is defended by S, which contradicts the assumption that S is a complete extension for SF. The same argument applies for the other direction.

(g) For $\sigma = grd$:
Follows from the fact that the complete extensions for SF and SF' coincide (apart from a^*).

(h) For $\sigma = stg$:
We have that $S \in cf(SF)$ iff $S' \in cf(SF')$. Assume $S \in stg(SF)$ and there is a conflict-free set $T \subseteq A \cup \{a^*\}$ such that $T^{+}_{R'} \supset S'^{\oplus}_{R'}$. Then $T^{\oplus}_R \supset S^{\oplus}_R$, which contradicts the assumption that S is a stage extension for SF. The same argument applies for the other direction.

(i) For $\sigma = sem$:
We have that $S \in adm(SF)$ iff $S' \in adm(SF')$. Assume $S \in sem(SF)$ and there is an admissible set $T \subseteq A \cup \{a^*\}$ such that $T^{\oplus}_{R'} \supset S'^{\oplus}_{R'}$. Then $T^{\oplus}_R \supset S^{\oplus}_R$, which contradicts the assumption that S is a semi-stable extension for SF. The same argument applies for the other direction.

□

Lemma 35. *Let $\sigma \in \{stb, pref, com, grd, sem\}$ and let $SF = (A, R)$ be a SETAF with $SF' = Tr_4(SF) = (A', R')$. Then each extension $E \in \sigma(SF')$ contains the argument that is added by the translation, i.e. for each $E \in \sigma(SF')$ we have $a^* \in E$.*

Proof. a^* is in $F(\emptyset)$ (where F is the characteristic function), and therefore in $grd(SF')$. Hence, it is in every extension in $com(SF')$, $pref(SF')$, $stb(SF')$, and $sem(SF')$, since all of these extensions contain the grounded extension. □

Note that Lemma 35 does not hold for the semantics *cf*, *adm*, *naive*, and *stg*. The empty set is conflict-free and admissible, and for each attack $(T, h) \in SF$ there is a conflict-free set $S = T \cup \{h\}$ in $SF' = Tr_4(SF)$. Therefore there are naive extensions $S' \supseteq S$ of SF' that do not contain the newly added argument. Moreover if some SETAF $SF = (A, R)$ has no stable extension, then the set A is conflict-free in SF', and there is no set $S \subseteq A^{SF'}$ such that $S^{\oplus}_{R'} \supset A^{\oplus}_{R'}$ in SF' (any set containing a^* cannot reach all original arguments, as this would imply the existence of a stable extension in SF).
These results are already sufficient to prove the full complexity of γ'-acyclic SETAFs (and therefore also for γ- and β-acyclic SETAFs).

Theorem 13. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the problems $Cred_\sigma$ and $Skept_\sigma$ for γ'-acyclic SETAFs have their full complexity, i.e. the complexity results in Table 3.7 hold.*

Proof. The respective memberships follow from the general case for SETAFs.
For hardness of the semantics *sem*, *stb*, *pref*, *com*, and *grd* it suffices to note that the respective problems are hard for SETAFs SF with $mincl(SF) \geq 3$ (by instancing Corollary 4) and that for a SETAF SF with $mincl(SF) \geq 3$ its translation $Tr_4(SF)$ is always γ'-acyclic. The NP-hardness for $Cred_{adm}$ for γ'-acyclic SETAFs follows from the identity $Cred_{adm} = Cred_{com}$. $Cred_{stg}$ and $Skept_{stg}$ are Σ_2^P-hard and Π_2^P-hard respectively even for AFs with only a single cycle of length 1 (see Reduction 2), these AFs are also γ'-acyclic, as they have no cycles of length ≥ 2. □

Since Reduction 2 we used to prove the hardness of γ'-acyclic SETAFs for the stage-semantics is strong-attack-cyclic, we need to slightly adapt it to transform it into a strong-attack-cycle-free SETAF.
The 'purpose' of the argument with the self-loop in Reduction 2 is only to ensure incomparability of the range of different stage extensions: let m be the argument with the self-loop (b in the reduction) and l an argument that attacks m (φ in the reduction). The only way to have $m \in E^{\oplus}_R$ for some stage extension E is to have $l \in E$. We can get the same behaviour by replacing m by some SETAF SF^*, such that $stb(SF^*) = \emptyset$ and letting l attack all arguments in SF^*. Since it has no stable extensions, the only way to get all arguments of SF^* into the range of a stage extension is again to have l in that extension.
If we take a strong-attack-cycle-free SETAF for SF^*, we can prove the hardness for stage semantics for strong-attack-cycle-free SETAFs. For this purpose we can take the SETAF from Example 3 (see Figure 3.10).

Example 3. *$SF^* = (A^*, R^*)$, where $A^* = \{w, x, y, z\}$, and $R^* = \{(\{w, x\}, y), (\{w, y\}, z), (z, x)\}$.*
We have $stb(SF^*) = \emptyset$ and $stg(SF^*) = \{\{w, x\}, \{w, y\}, \{w, z\}, \{y, z\}\}$.

Lemma 36. *Let $SF = (A, R)$ be a SETAF and let $l, m \in A$ be arguments such that $\{(l, m), (m, m)\} \subseteq R$, and no further attacks involve m, i.e. there is no other attack*

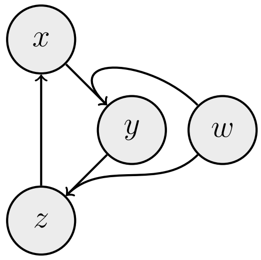


Figure 3.10: Strong-attack-cycle-free SETAF SF^* from Example 3 with no stable extensions.

$(T, h) \in R$ such that $m \in T$ or $m = h$. Furthermore let SF^* be the SETAF from Example 3.
Then for the SETAF $SF' = (A', R')$, where

$$\begin{aligned} A' &= (A \cup A^{SF^*}) \setminus \{m\} \\ R' &= (R \setminus \{(l, m), (m, m)\}) \cup R^{SF^*} \cup \{(l, x) \mid x \in A^{SF^*}\} \end{aligned}$$

we have that $stg(SF) = \{E' \cap A \mid E' \in stg(SF')\}$.

Proof. "$\subseteq$": We show that if $E \in stg(SF)$ then there is some $E' \subseteq A'$ with $E' \supseteq E$ such that $E' \in stg(SF')$. Assume $E \in stg(SF)$. If $l \in E$, then $E' = E \in stg(SF')$; if $l \notin E$, then $E \cup \{y, z\} \in stg(SF')$: in both cases it is clear that E' is conflict-free in SF', and any set $S' \subseteq A'$ with $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$ would yield a set $S = S' \cap A$, such that $S \in cf(SF)$ with $S^{\oplus}_{R} \supset E^{\oplus}_{R}$, which would be a contradiction to the assumption that E is stage in SF, so there cannot be such a set S', hence, E' is stage in SF'.
"$\supseteq$": We show that if $E' \in stg(SF')$ then $E' \cap A \in stg(SF)$. Assume $E' \in stg(SF')$. If $l \in E'$, then $E' \cap A = E' \in stg(SF)$; if $l \notin E'$, then still $E' \cup A \in stg(SF')$: in both cases it is clear that E is conflict-free in SF, and any set $S \subseteq A$ with $S^{\oplus}_{R} \supset E^{\oplus}_{R}$ would yield a set $S' = S \cup \{y, z\}$, such that $S' \in cf(SF')$ with $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$, which would be a contradiction to the assumption that E' is stage in SF', so there cannot be such a set S, hence, E is stage in SF. □

Theorem 14. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the problems $Cred_\sigma$ and $Skept_\sigma$ for strong-attack-cycle-free SETAFs have their full complexity, i.e. the complexity results in Table 3.7 hold.*

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	P-c	P-c	Π_2^P-c	Π_2^P-c

Table 3.7: The complexity for strong-attack-cycle-free, β-acyclic, γ-acyclic, and γ'-acyclic SETAFs coincides with the full complexity for general SETAFs.

Proof. The respective memberships follow from the general case for SETAFs.
For hardness of the semantics *sem*, *stb*, *pref*, *com*, and *grd* it suffices to note that for every AF $F = (A, R)$ its translation $Tr_4(F)$ is a strong-attack-cycle-free SETAF where an argument $a \in A$ is credulously/skeptically accepted iff a is accepted credulously/skeptically in F. The NP-hardness for $Cred_{adm}$ for strong-attack-acyclic SETAFs follows the identity $Cred_{adm} = Cred_{com}$. $Cred_{stg}$ and $Skept_{stg}$ are Σ_2^P-hard and Π_2^P-hard respectively even for AFs with only a single cycle of length 1 (see Reduction 2), by further applying the construction from Lemma 36 with $\varphi = l$ and $b = m$ we get the hardness for strong-attack-cycle-free SETAFs. □

3.5 Summary

In this chapter we discovered that acyclicity and even-cycle-freeness are tractable fragments for SETAFs w.r.t. all semantics under our consideration. Moreover we discovered another tractable fragment for AFs and SETAFs w.r.t. all semantics but *stg*, that is, frameworks with a maximum cycle length of 1 (this subsumes acyclicity). In fact, we found out that there is a lower complexity for certain semantics up to a maximum cycle length of 3, SETAFs with longer cycles have their full complexity. A minimum cycle length, on the other hand, turned out not to lower the complexity. We investigated the effect of odd-cycle-freeness on SETAFs, which lowers the complexity to the first level of the polynomial hierarchy, just as in AFs.

We established that most adaptations of acyclicity-notions from the hypergraph literature do not yield a lower complexity in general. In particular, we adapted Berge-acyclicity, which complexity-wise coincides with even-cycle-freeness, and β-, γ-, and δ-acyclicity, which turned out to have the full complexity. Attempts to generalize α-acyclicity to SETAFs proved fruitless, as an adaptation would not generalize acyclicity of directed graphs.

Finally, we suggested a new notion of cycles, the (strong)-attack-cycles. When it comes to the complexity of reasoning in strong-attack-acyclic SETAFs, we established there is no decrease in the complexity either. The results from this chapter are summarized in Table 3.8.

		adm	*stb*	*pref*	*com*	*stg*	*sem*
Acyclic SETAFs,	$Cred_\sigma$	P-c	P-c	P-c	P-c	P-c	P-c
$maxcycle(SF) = 0$	$Skept_\sigma$	trivial	P-c	P-c	P-c	P-c	P-c
Even-cycle-free, Berge-	$Cred_\sigma$	P-c	P-c	P-c	P-c	Σ_2^P-c	P-c
acyclic, δ-acyclic SETAFs	$Skept_\sigma$	trivial	P-c	P-c	P-c	Π_2^P-c	P-c
Odd-cycle-free	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	NP-c	NP-c
SETAFs	$Skept_\sigma$	trivial	coNP-c	coNP-c	P-c	coNP-c	coNP-c
Short-cycle-free	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
SETAFs	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	P-c	Π_2^P-c	Π_2^P-c
$maxcycle(SF) \leq 1$	$Cred_\sigma$	P-c	P-c	P-c	P-c	Σ_2^P-c	P-c
	$Skept_\sigma$	trivial	P-c	P-c	P-c	Π_2^P-c	P-c
$2 \leq maxcycle(SF) \leq 3$	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	coNP-c	coNP-c	P-c	Π_2^P-c	Π_2^P-c
$maxcycle(SF) \geq 4$	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	P-c	Π_2^P-c	Π_2^P-c
strong-attack-cycle-free,	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
β-, γ-, γ'-acyclic SETAFs	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	P-c	Π_2^P-c	Π_2^P-c

Table 3.8: Summary of the complexity results of chapter 3. The complexity of *cf* and *naive* semantics have been omitted, as they are in L, and the complexity of *grd* was omitted, as it is P-complete for all problems in this table.

CHAPTER 4

Symmetry

In this chapter we will discuss different notions of symmetry for SETAFs. In Section 4.1 we will recapitulate existing results for self-attack-free symmetric AFs and symmetric AFs allowing self-attacks, and state some properties thereof. After that in Section 4.2 we will define different notions of symmetry for SETAFs; in particular we will generalize properties of symmetric AFs and show which of them can be used for symmetry-definitions on directed hypergraphs that properly generalize classical symmetry.

Finally, in Section 4.3 we will examine the complexity of reasoning problems in SETAFs that have the respective symmetric properties, each both for self-attack-free SETAFs and for SETAFs allowing self-attacks.

4.1 State of the Art in AFs

A relation R is symmetric iff for every pair $(a, b) \in R$ there is a corresponding "symmetric" pair $(b, a) \in R$; AFs with symmetric attack relations are called *symmetric* AFs, or, to avoid confusion with other notions of symmetry in the following, *AF-symmetric*. This means every argument defends itself against all attacks towards it, in particular it means conflict-free sets coincide with admissible sets, naive extensions coincide with preferred extensions, and stage extensions coincide with semi-stable extensions. This will not hold for all generalizations we will encounter throughout this chapter.

The complexity of symmetric AFs has first been studied with the additional condition that they are self-attack-free [CMDM05] (in which case in most semantics every single argument is contained in at least one extension, which is why credulous acceptance is trivially true); then this has been generalized to symmetric AFs allowing self-attacks [Dvo12a]. Both properties allow us to reason more efficiently, in particular the complexity of the reasoning problems w.r.t. the semantics under our consideration is given in Ta-

Self-attack-free symmetric AFs

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	trivial	trivial	trivial	trivial	trivial	in L	trivial	trivial
$Skept_\sigma$	trivial	trivial	in L	in L	in L	in L	in L	in L	in L

Symmetric AFs allowing self-attacks

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	in L	in L	NP-c	in L	in L	in L	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	in L	in L	in L	Π_2^P-c	Π_2^P-c

Table 4.1: Complexity for self-attack-free symmetric AFs and symmetric AFs allowing self-attacks.

ble 4.1. The identified tractable fragments are irreflexive (i.e. self-attack-free) symmetric AFs for all semantics under our consideration and symmetric AFs allowing self-attacks for $\sigma \in \{cf, adm, naive, pref, com\}$.

4.2 Symmetry in SETAFs

In the following we will examine and generalize properties of symmetric AFs. Some of these properties will turn out to be unfruitful when it comes to defining symmetry for SETAFs, these efforts are stated in the following. After that we will provide proper definitions for symmetric SETAFs.

4.2.1 Unsuitable Definitions

On AFs, we can characterize symmetry with the following condition: an AF is AF-symmetric iff for all arguments a and b we have $a \rightarrowtail b \Leftrightarrow b \rightarrowtail a$. Intuitively, we could adapt this notion for collective attacks by generalizing arguments to sets. However, for redundancy-free SETAFs this notion coincides with AF-symmetry.

Definition 33. *A SETAF $SF = (A, R)$ is α-symmetric iff for all sets $T, H \subseteq A$ we have that $T \rightarrowtail H \Leftrightarrow H \rightarrowtail T$.*

Lemma 37. *Every redundancy-free α-symmetric SETAF is an AF.*

Proof. Assume towards contradiction there is a redundancy-free α-symmetric SETAF $SF = (A, R)$ that is not an AF, i.e. there is an attack $(T, h) \in R$ such that $|T| > 1$. Since $T \rightarrowtail \{h\}$ and because SF is α-symmetric we have $\{h\} \rightarrowtail T$, i.e. there is an attack $(\{h\}, t) \in R$ where $t \in T$. Now we have $\{h\} \rightarrowtail \{t\}$, and hence by α-symmetry $\{t\} \rightarrowtail \{h\}$, which means there is an attack $(\{t\}, h) \in R$. Since $\{t\} \subseteq T$ the attack (T, h) is redundant, which contradicts the assumption that SF was redundancy-free. ☐

Likewise we immediately get to the following result.

Lemma 38. *Let $SF = (A, R)$ be a redundancy-free SETAF. Then SF is α-symmetric iff it is AF-symmetric.*

By relaxing this notion we can get another characterization. We will see that the next definition is also not useful, as it does not generalize symmetry.

Definition 34. *A SETAF $SF = (A, R)$ is β-symmetric iff for all sets $T, H \subseteq A$ we have that $T \rightarrowtail H \Rightarrow$ there is a set $H' \supseteq H$ such that $H' \rightarrowtail T$.*

We have that the AF in Figure 4.1a is β-symmetric, as the set $\{a, b, c\}$ attacks all arguments, but it is clearly not classically symmetric.

4.2.2 Suitable Definitions

Another equivalent way to characterize symmetry on AFs is with the following condition: an AF is AF-symmetric iff for every attack (a, b) there is an attack (b, a). This yields different definitions of symmetry for SETAFs. For an illustrating examples see Figure 4.1b and Figure 4.1c, respectively. The latter notion is equivalent to the definition of symmetric SETAFs as given in [DKZLW20, p. 214].

Definition 35. *A SETAF $SF = (A, R)$ is γ-symmetric iff for every attack $(T, h) \in R$ there is an attack $(H, t) \in R$ with $h \in H$ for* some *$t \in T$.*

Definition 36. *A SETAF $SF = (A, R)$ is δ-symmetric iff for every attack $(T, h) \in R$ there is an attack $(H, t) \in R$ with $h \in H$ for* all *$t \in T$.*

Note that a SETAF is δ-symmetric iff its primal graph is symmetric. Finally, we will provide a very restricting notion of symmetry for SETAFs. For an illustrating example see Figure 4.1d.

Definition 37. *A SETAF $SF = (A, R)$ is ϵ-symmetric iff for every attack $(T, h) \in R$ we either have*

- *$T = \{h\}$, or*
- *if $T \neq \{h\}$ and $h \in T$, there are attacks such that $\forall x \in T$ we have $(T, x) \in R$, or*
- *if $h \notin T$, there are attacks such that $\forall x \in S$ we have $(S \setminus \{x\}, x) \in R$ where $S = T \cup \{h\}$.*

We have that γ-symmetry, δ-symmetry, and ϵ-symmetry coincide with classical symmetry on AFs. Moreover every ϵ-symmetric SETAF is δ-symmetric, every δ-symmetric SETAF is γ-symmetric, but the converse does not hold, as Figure 4.1b and Figure 4.1c illustrates.

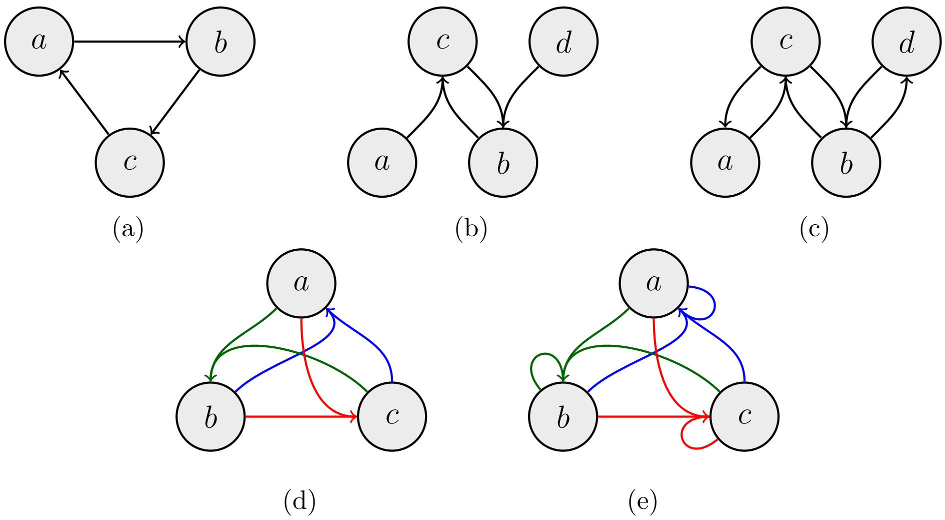


Figure 4.1: Illustration of the different symmetry notions: (a) a β-symmetric AF, (b) a γ-symmetric SETAF that is not δ-symmetric, (c) a δ-symmetric SETAF that is not ϵ-symmetric, and (d) and (e) two ϵ-symmetric SETAFs.

4.3 Complexity

In the following we will investigate the complexity of our reasoning problems for the defined symmetric classes of SETAFs. First we will deal with γ- and δ-symmetry at once and exploit the fact that the complexity of δ-symmetric SETAFs is a lower bound for the complexity of γ-symmetric SETAFs, while the complexity of γ-symmetric SETAFs is an upper bound for the complexity of δ-symmetric SETAFs. Then we will examine the complexity of ϵ-symmetric SETAFs.

4.3.1 γ- and δ-symmetry

We start by providing a translation for $\sigma \in \{cf, adm, stb, pref, stg, sem\}$ such that for every self-attack-free SETAF SF its translation is δ-symmetric. Moreover we will establish that this translation is efficient and acceptance-preserving. Note that we use Tr_5 only for self-attack-free SETAFs. For an illustration of Tr_5 see Figure 4.2.

Translation 5. *Let $SF = (A, R)$ be a SETAF. The SETAF-translation Tr_5 is defined as $Tr_5(SF) = (A', R')$ with*

$$\begin{aligned} A' &= A \cup \{a^1_{r,t}, a^2_{r,t} \mid r = (T, h) \in R, t \in T\}, \\ R' &= R \cup \{(a^1_{r,t}, a^2_{r,t}), (a^2_{r,t}, a^1_{r,t}), (\{a^1_{r,t}, a^2_{r,t}, h\}, t), (t, a^1_{r,t}), (t, a^2_{r,t}) \mid r = (T, h) \in R, t \in T\} \end{aligned}$$

Intuitively, for each attack in the original SETAF the translation Tr_5 adds an attack towards every attacker. Since the arguments a^1_r and a^2_r attack each other, this added attack is inactive, and, hence, most semantics do not change. Also in order to preserve

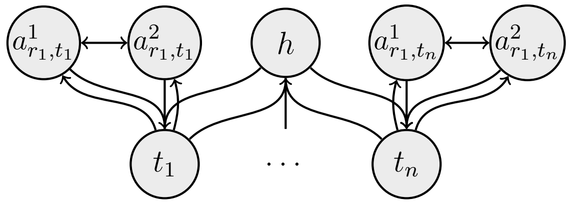


Figure 4.2: Illustration Translation Tr_5. We have a SETAF with one attack $r_1 = (\{t_1, \dots, t_n\}, h)$.

admissibility we add an attack towards the added arguments. We have that Tr_5 is efficient and embedding.

Lemma 39. *Let $SF = (A, R)$ be a SETAF and let $SF' = (A', R') = Tr_5(SF)$. Then for every $E' \in cf(SF')$ we have for $E = E' \cap A$ that $E^{\oplus}_R = E'^{\oplus}_{R'} \cap A$.*

Proof. "$\subseteq$": Immediate by the fact that Tr_5 is embedding and the monotonicity of $(.)^{\oplus}$. "$\supseteq$": Note that the set of active attacks towards arguments in A in SF' is the set of active attacks in SF. The only active attacks towards arguments in A in SF' are from within A. The fact that in the construction of SF' no further attacks between arguments in A is added concludes the proof. □

This brings us to our first result that will allow us to settle the complexity of reasoning for $\sigma \in \{cf, adm, stb, pref, stg, sem\}$ in self-attack-free γ- and δ-symmetric SETAFs.

Theorem 15. *Let $\sigma \in \{cf, adm, stb, pref, stg, sem\}$. Then Tr_5 is an acceptance-preserving translation for $\sigma \Rightarrow \sigma$ such that for every self-attack-free SETAF SF its translation $SF' = Tr_5(SF) = (A', R')$ is δ-symmetric.*

Proof. We will show two statements for each semantics σ: firstly we will show constructively that for any extension $E \in \sigma(SF)$ there exists an extension $E' \in \sigma(SF')$ such that $E' \cap A = E$ ("$\Rightarrow$"). Secondly we will show that for each extension $E' \in \sigma(SF')$ the corresponding extension $E = E' \cap A$ is an extension $E \in \sigma(SF)$ ("$\Leftarrow$").

(a) For $\sigma = cf$:
"$\Rightarrow$": Let $E \in cf(SF)$. Then also $E \in cf(SF')$, as there are no attacks between elements of A that are added in the construction.
"$\Leftarrow$": Let $E' \in cf(SF)$ and let $E = E' \cap A$. Then $E \in cf(SF)$, as there can be no attack between arguments in A.

(b) For $\sigma = adm$:
"$\Rightarrow$": Let $E \in adm(SF)$ and let $E' = E \cup \{a^1_{r,t} \mid r = (T, h), t \in T, E \rightarrowtail_R t\}$. By construction we have $E' \in cf(SF')$. Assume towards contradiction some $a \in E'$ is not defended by E', i.e. there is an attack $(T, a) \in R'$ such that $E' \not\rightarrowtail_{R'} T$.

This means either $a \in A' \setminus A$ or $a \in A$. In the first case we have $a = a^1_{r,t}$ for some $r = (T, h) \in R$ with $t \in T$. We have that a defends itself against the attack from $a^2_{r,t}$, the only remaining attack towards a is from t. But since $a \in E'$, by construction we have $E \rightarrowtail_R t$, which also means $E' \rightarrowtail_{R'} t$, so a is defended by E', which is a contradiction. In the second case we have $a \in A$. Since $a \in E$ and $E \in adm(SF)$ we know that a is defended against all attacks in R, i.e. all attacks from within A. But since the only active attacks towards a are from within A, we have that a is defended, which is a contradiction.
"$\Leftarrow$": Let $E' \in adm(SF')$ and let $E = E' \cap A$. We know $E \in cf(SF)$. Let $a \in E$ and let $(T, a) \in R$ be an attack towards a. Since E' is admissible in SF' we have $E' \rightarrowtail_{R'} T$, i.e. there is an attack $(T', t) \in R'$ such that $t \in T$ and $T' \subseteq E'$. Since the only active attacks towards t are from within A, we also have that $E \rightarrowtail_R t$, which means a is defended by E in SF.

(c) For $\sigma = stb$:
"$\Rightarrow$": Let $E \in stb(SF)$ and let $E' = \{a^1_{r,t} \mid r = (T, h), t \in T, t \notin E\}$. We have $E \in cf(SF')$ by construction. Moreover, since $E \in stb(SF)$, by Lemma 39 we have $A \subseteq E'^{\oplus}_{R'}$, and by construction we have $A' \setminus A \subseteq E'^{\oplus}_{R'}$.
"$\Leftarrow$": Let $E' \in stb(SF')$ and let $E = E' \cap A$. We know $E \in cf(SF)$, and, since $E' \in stb(SF')$, by Lemma 39 we have $A \subseteq E^{\oplus}_{R}$.

(d) For $\sigma = pref$:
"$\Rightarrow$": Let $E \in pref(SF)$ and let $E' = \{a^1_{r,t} \mid r = (T, h), t \in T, E \rightarrowtail_R t\}$. We already know $E' \in adm(SF')$. Assume towards contradiction there is a set $S' \in adm(SF')$ such that $S' \supset E'$, i.e. there is an argument $a \in A'$ such that $a \in S' \setminus E'$. This means either $a \in A$ or $a \in A' \setminus A$. Let $S = S' \cap A$, we know $S \in adm(SF)$. In the first case we would have $S \supset E$, which is a contradiction to the assumption that $S \in pref(SF)$. In the second case we have $a \in A' \setminus A$, i.e. $a = a^1_{r,t}$ (or $a = a^2_{r,t}$, in which case the proof continues analogously) for some $r = (T, h) \in R$ and $t \in T$. Since a is attacked by t, in order to defend it we have $S' \rightarrowtail_{R'} t$. Since the only active attacks towards t are from within A, there must be an attack $(T', t) \in R$ such that $T' \subseteq S'$. We know $E' \not\rightarrowtail_{R'} t$ by construction, so there is an argument $b \in A$ such that $b \in S' \setminus E'$, but since $S \in adm(SF)$ and $S \supset E$ again we have a contradiction to the assumption that $S \in pref(SF)$.
"$\Leftarrow$": Let $E' \in pref(SF')$ and let $E = E' \cap A$. We know $E \in adm(SF)$. Assume towards contradiction there is a set $S \in adm(SF)$ such that $S \supset E$. Let $S' = S \cup (E' \setminus E)$. By construction we have $S' \supset E'$. Moreover we have $S' \in adm(SF')$: assume towards contradiction there is an argument $a \in S'$ that is not defended by S', i.e. there is an attack $(T, a) \in R'$ such that $S' \not\rightarrowtail_{R'} T$. We either have $a \in A$ or $a \in A' \setminus A$. In the first case a defends itself against attacks from $A' \setminus A$, and it is defended against attacks from A, since $a \in S$ and $S \in adm(SF)$. In the second case we have $a = a^1_{r,t}$ (or $a = a^2_{r,t}$, in which case the proof continues analogously) for some $r = (T, h) \in R$ and $t \in T$. We have that a defends itself against the attack from $a^2_{r,t}$. It is also attacked from t, but we have $S' \rightarrowtail_{R'} t$: since $a \in S'$

and $a \in A' \setminus A$ by construction of S' we have $a \in E'$, but since $E' \in adm(SF')$ we have $E' \rightarrowtail_{R'} t$. The argument t can only be actively attacked from within A (since there are no other active attacks towards t in R') and, hence, $S' \rightarrowtail_{R'} t$. This shows $S' \in adm(SF')$, and since $S' \supset E'$ we have a contradiction to the assumption $E' \in pref(SF')$.

(e) For $\sigma = stg$:
"⇒": Let $E \in stg(SF)$ and let $E' = \{a^1_{r,t} \mid r = (T,h), t \in T, t \notin E\}$. We have $E' \in cf(SF')$ by construction. Assume towards contradiction there is a set $S' \in cf(SF')$ such that $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$. Let $S = S' \cap A$. We know $S \in cf(SF)$. Moreover we have $S^{\oplus}_R \supseteq E^{\oplus}_R$ by Lemma 39. $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$ means there is an argument $a \in S'^{\oplus}_{R'} \setminus E'^{\oplus}_{R'}$. This means either $a \in A$ or $a \in A' \setminus A$. Since we have $A' \setminus A \subseteq S'^{\oplus}_{R'}$ by construction, the second option is impossible. So there is an argument $a \in A$ such that $a \in S'^{\oplus}_{R'} \setminus E'^{\oplus}_{R'}$, but then $a \in S^{\oplus}_R \setminus E^{\oplus}_R$, so $S^{\oplus} \supset E^{\oplus}$, which is a contradiction to our assumption $E \in stg(SF)$.
"⇐": Let $E' \in stg(SF')$ and let $E = E' \cap A$. We know $E \in cf(SF)$. Assume towards contradiction there is a set $S \in cf(SF)$ such that $S^{\oplus}_R \supset E^{\oplus}_R$. Let $S' = \{a^1_{r,t} \mid r = (T,h), t \in T, t \notin S\}$. We have $S' \in cf(SF')$ by construction. As before we have $A' \setminus A \subseteq S'^{\oplus}_{R'}$. Moreover, by Lemma 39 we have $S'^{\oplus}_{R'} \cap A \supset E'^{\oplus}_{R'} \cap A$, so we have $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$, which is a contradiction to the assumption $E' \in stg(SF')$.

(f) For $\sigma = sem$:
"⇒": Let $E \in sem(SF)$ and let $E' = \{a^1_{r,t} \mid r = (T,h), t \in T, E \rightarrowtail_R t\}$. We already know $E' \in adm(SF')$. Assume towards contradiction there is a set $S' \in adm(SF')$ such that $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$. Let $S = S' \cap A$. We know $S \in adm(SF)$. Moreover by Lemma 39 we have $S^{\oplus}_R \supseteq E^{\oplus}_R$. From $S'^{\oplus}_{R'} \supset E'^{\oplus}_{R'}$ we know there is an argument $a \in A'$ such that $a \in S'^{\oplus}_{R'}$ but $a \notin E'^{\oplus}_{R'}$. This means either $a \in A$ or $a \in A' \setminus A$. In the first case by Lemma 39 we get $S^{\oplus}_R \supset E^{\oplus}_R$, which is a contradiction to our assumption $E \in sem(SF)$. In the second case we have $a = a^1_{r,t}$ (or $a = a^2_{r,t}$, in which case the proof continues analogously) for some $r = (T,h) \in R$ and $t \in T$. We have $S' \rightarrowtail_{R'} t$ in order to defend a. But by construction of E' we have $E' \not\rightarrowtail_{R'} t$, hence, $E \not\rightarrowtail_R t$, but since $S \rightarrowtail_R t$ we have $S^{\oplus}_R \supset E^{\oplus}_R$, which is a contradiction to our assumption $E \in sem(SF)$.
"⇐": Let $E' \in sem(SF')$ and let $E = E' \cap A$. We know $E \in adm(SF)$. Assume towards contradiction there is a set $S \in adm(SF)$ such that $S^{\oplus}_R \supset E^{\oplus}_R$. Let $S' = \{a^1_{r,t} \mid r = (T,h), t \in T, S \rightarrowtail_R t\}$. By construction we have $S' \in adm(SF')$. By Lemma 39 we have $S'^{\oplus}_{R'} \cap A \supseteq E'^{\oplus}_{R'} \cap A$. Moreover we have $S'^{\oplus}_{R'} \cap A' \setminus A \supseteq E'^{\oplus}_{R'} \cap A' \setminus A$: Assume otherwise, i.e. there is an argument $a \in A' \setminus A$ such that $a \in E'^{\oplus}_{R'} \setminus S'^{\oplus}_{R'}$. We have $a = a^1_{r,t}$ (or $a = a^2_{r,t}$, in which case the proof continues analogously) for some $r = (T,h) \in R$ and $t \in T$. We either have $a \in E'$ or $t \in E'$. In the first case in order to defend a we would have $E' \rightarrowtail_{R'} t$. The argument t can only be attacked from within A, so we would also have $S \rightarrowtail_R t$ and, hence, $S' \rightarrowtail_{R'} t$, which means $a \in S'^{\oplus}_{R'}$, which is a contradiction. In the second case we have $t \in E'$, which means

$t \in E_R^{\oplus}$, so by assumption $t \in S_R^{\oplus}$, and then again by construction $a \in S'^{\oplus}_{R'}$ (either because $t \in S'$ or because $S \rightarrowtail_R t$).

□

In the following we will show that deciding whether an argument is in the grounded extension of a γ-symmetric SETAF is doable efficiently, namely in L. As we do not further restrict the γ-symmetric SETAFs this then carries over to δ-symmetric SETAFs and to self-attack-free γ- and δ-symmetric SETAFs.

Lemma 40. *Let $SF = (A, R)$ be a γ-symmetric SETAF. Then an argument $a \in A$ is in the grounded extension G iff a is not in the head of any attack, i.e. there is no attack $(T, a) \in R$.*

Proof. Consider the construction of G with the characteristic function $\mathcal{F}_{SF}$. In the first step, exactly the arguments that are not in the head of an attack are added (which concludes the "$\Leftarrow$"-direction).
Now there are no arguments left that are defended by $\mathcal{F}_{SF}(\emptyset)$: towards contradiction assume there is an argument a that is defended by $\mathcal{F}_{SF}(\emptyset)$, but not in it. There is at least one attack $(T, a) \in R$, otherwise a would be in $\mathcal{F}_{SF}(\emptyset)$. In order to defend a we would have $\mathcal{F}_{SF}(\emptyset) \rightarrowtail_R T$, i.e. there is an attack (T', t) with $T' \subseteq \mathcal{F}_{SF}(\emptyset)$ and $t \in T$. But since SF is γ-symmetric there is another attack $(T'', e) \in R$ such that $t \in T''$ and $e \in T'$, which is a contradiction, since the arguments in $\mathcal{F}_{SF}(\emptyset)$ are not in the head of any attack. This means $\mathcal{F}_{SF}(\emptyset) = \mathcal{F}_{SF}(\mathcal{F}_{SF}(\emptyset))$, which concludes the "$\Rightarrow$"-direction. □

Summarizing the previous results we have the full complexity landscape for γ- and δ-symmetric SETAFs.

Theorem 16. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ for the problems $Cred_\sigma$ and $Skept_\sigma$ of redundancy-free self-attack-free γ- and δ-symmetric SETAFs the complexity results in Table 4.2 hold.*

Proof. The memberships of the respective problems follow from the respective results for arbitrary SETAFs. Note that, as we have no self-attacks, for every argument $a \in A$ the set $\{a\}$ is conflict-free and, hence, part of a naive extension. By Theorem 15 we get the respective hardness results for $Cred_\sigma$ and $Skept_\sigma$ for $\sigma \in \{adm, stb, pref, stg, sem\}$. The hardness of $Cred_{com}$ follows from the identity $Cred_{com} = Cred_{adm}$. Finally, the L-membership of $Cred_{grd}$, $Skept_{grd}$, and $Skept_{com}$ follows from Lemma 40. □

Likewise we obtain the following result for general (i.e. not necessarily self-attack-free) SETAFs. Note that, since self-attacks are allowed, $Cred_{cf}$ and $Cred_{naive}$ are in L, as an argument is credulously accepted iff it is not in a self-loop, which can easily be checked in L.

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	NP-c	trivial	NP-c	NP-c	NP-c	in L	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	in L	in L	Π_2^P-c	Π_2^P-c

Table 4.2: The complexity for redundancy-free self-attack-free γ- and δ-symmetric SETAFs. The trivial results in the first line are 'yes'-instances, the trivial results in the second line are 'no'-instances.

Theorem 17. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ for the problems $Cred_\sigma$ and $Skept_\sigma$ of redundancy-free γ- and δ-symmetric SETAFs the complexity results in Table 4.3 hold.*

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	in L	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	in L	in L	Π_2^P-c	Π_2^P-c

Table 4.3: The complexity for redundancy-free γ- and δ-symmetric SETAFs allowing self-attacks.

4.3.2 ϵ-symmetry

As with γ- and δ-symmetric SETAFs we will examine the complexity of self-attack-free ϵ-symmetric SETAFs and ϵ-symmetric SETAFs allowing self-attacks separately. Here we do not have that every conflict-free set is admissible (as it is the case in symmetric AFs), which is illustrated by Figure 4.1e: we have that $\{a, b\}$ is conflict-free, but not admissible.

Allowing self-attacks

First we will show that reasoning on the grounded extension is efficient, in particular we have that the following lemma allows us to decide our reasoning problems w.r.t. *grd* semantics in L.

Lemma 41. *Let $SF = (A, R)$ be an ϵ-symmetric SETAF. Then an argument $a \in A$ is in the grounded extension G of SF iff it is not involved in any attack.*

Proof. For the $\Rightarrow$-direction consider the construction of G via the characteristic function $\mathcal{F}_{SF}$. In the first step, exactly the arguments that are not involved in any attacks are added. As every other argument is now attacked (i.e. not defended by $\mathcal{F}_{SF}(\emptyset)$), a fix point is reached. The $\Leftarrow$-direction follows from the definition of the grounded extension. □

To show that reasoning w.r.t. *adm* semantics has the full complexity in ϵ-symmetric SETAFs, consider the following ϵ-symmetric variation of the standard reduction. For an illustration of the next reduction see Figure 4.3.

Reduction 5. *Let φ be a CNF-formula with a set of clauses C over propositional atoms Y. We define $SF^5_\varphi = (A', R')$, where*

$$\begin{aligned} A' &= \{\varphi\} \cup C \cup Y \cup \bar{Y} \\ R' &= \{(\{c, \varphi\}, \varphi), (\{c, \varphi\}, c) \mid c \in C\} \cup \{(y, \bar{y}), (\bar{y}, y) \mid y \in Y\} \cup \\ &\quad \{(y, c), (c, y) \mid y \in c, c \in C\} \cup \{(\bar{y}, c), (c, \bar{y}) \mid \bar{y} \in c, c \in C\} \end{aligned}$$

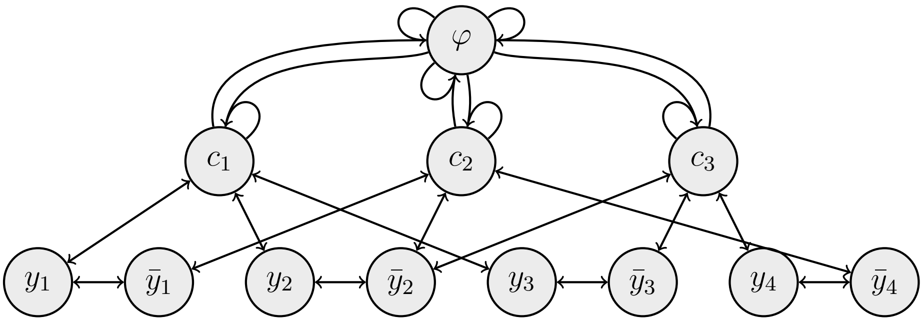


Figure 4.3: Illustration of SF^5_φ for a formula φ with $Y = \{y_1, y_2, y_3, y_4\}$, and $C = \{\{y_1, y_2, y_3\}, \{\bar{y}_1, \bar{y}_2, \bar{y}_4)\}, \{\bar{y}_2, \bar{y}_3, y_4\}\}$.

The only changes to the standard reduction are some additional attacks in order to make the SETAF ϵ-symmetric, and that the attacks between φ and arguments $c \in C$ are now self-attacks. These attacks between φ and the arguments in C are always inactive, which means φ cannot defend itself against the attacks from the arguments in C. This means φ can only be in an admissible set, if all $c \in C$ are attacked by arguments $y \in Y$ and $\bar{y} \in \bar{Y}$, which lets us construct a satisfying truth assignment (see next lemma).

Lemma 42. *Let φ be a CNF-formula with a set of clauses C over propositional atoms Y. Then φ is satisfiable iff φ is credulously accepted in SF^5_φ w.r.t. adm semantics.*

Proof. First note that, as there can never be an active attack towards φ in SF^5_φ, no $c \in C$ can be defended against the attack from c and φ, therefore no such c can be in an admissible set.
"$\Rightarrow$": Assume φ is satisfiable, i.e. there is a truth assignment $\mathcal{I}$ such that $\mathcal{I} \models \varphi$. We can construct an admissible set in the following way: let $E = \{\varphi\} \cup \{y \mid y \in \mathcal{I}\} \cup \{\bar{y} \mid y \in Y \setminus \mathcal{I}\}$. E is conflict-free by construction. Moreover note that, as E was constructed from a satisfying assignment, each $c \in C$ is attacked, which means φ is defended against all attacks towards it, and also each $y, \bar{y} \in E$ is defended against the attacks from the arguments c. Finally, the arguments $y, \bar{y} \in E$ defend itself against the attack from their dual literal.
"$\Leftarrow$": Assume there is an admissible set $E \subseteq A'$ such that $\varphi \in E$. In order to defend φ we have $E \rightarrowtail_{R'} c$ for all $c \in C$. We have that for each $y \in Y$, at most one of y and $\bar{y}$

is in E. Now let $\mathcal{I}$ be an interpretation such that $y \in \mathcal{I} \Leftrightarrow y \in E$ for $y \in Y$. We have $\mathcal{I} \vDash \varphi$, as for each clause c at least one of its literals attacks c in E. □

To further show that also reasoning w.r.t. *pref* semantics has the full complexity in ϵ-symmetric SETAFs, consider the following ϵ-symmetric variation of the standard reduction to show Π_2^P-hardness for $Skept_{pref}$. For an illustration of the next reduction see Figure 4.4.

Reduction 6. *Let $\phi = \forall Y \exists Z C$ be a $QBF_\forall^2$ formula with sets of propositional atoms Y and Z and a conjunctive formula φ over a set of clauses C. We define $SF_\phi^6 = (A', R')$, where*

$$
\begin{aligned}
A' &= \{\varphi, \bar{\varphi}, \varphi'\} \cup C \cup Y \cup \bar{Y} \cup Z \cup \bar{Z} \\
R' &= \{(\{c, \varphi\}, \varphi), (\{c, \varphi\}, c) \mid c \in C\} \cup \{(y, \bar{y}), (\bar{y}, y) \mid y \in Y\} \ \cup \\
&\quad \{(y, c), (c, y) \mid y \in c, c \in C\} \cup \{(\bar{y}, c), (c, \bar{y}) \mid \bar{y} \in c, c \in C\} \ \cup \\
&\quad \{(\varphi, \bar{\varphi}), (\bar{\varphi}, \varphi), (\{\bar{\varphi}, \varphi'\}, \bar{\varphi}), (\{\bar{\varphi}, \varphi'\}, \varphi')\} \ \cup \\
&\quad \{(\{z, \bar{\varphi}\}, z), (\{z, \bar{\varphi}\}, \bar{\varphi}), (\{\bar{z}, \bar{\varphi}\}, \bar{z}), (\{\bar{z}, \bar{\varphi}\}, \bar{\varphi}) \mid z \in Z\}
\end{aligned}
$$

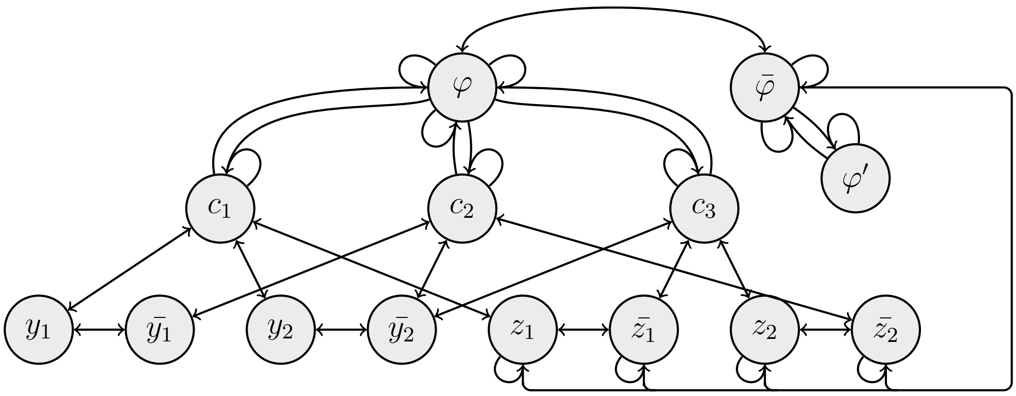


Figure 4.4: Illustration of SF_ϕ^6 for $\phi = \forall Y \exists Z \varphi(Y, Z)$ with $Y = \{y_1, y_2\}$, $Z = \{z_1, z_2\}$, and $C = \{\{y_1, y_2, z_1\}, \{\bar{y}_1, \bar{y}_2, \bar{z}_2)\}, \{\bar{y}_2, \bar{z}_1, z_2\}\}$. Note that the attacks between $\bar{\varphi}$ and $z \in Z$ (and $\bar{z} \in \bar{Z}$ respectively) are of the form $(\{\bar{\varphi}, z\}, z)$, $(\{\bar{\varphi}, z\}, \bar{\varphi})$, $(\{\bar{\varphi}, \bar{z}\}, \bar{z})$, and $(\{\bar{\varphi}, \bar{z}\}, \bar{\varphi})$ respectively, and overlay in this illustration only in the interest of presentability.

Similar to Reduction 5, in Reduction 6 we add additional attacks in order to make the SETAF ϵ-symmetric. As in the standard reduction for the Π_2^p-hardness of $Skept_{pref}$ (cf. [DD17, p. 22-24]) we have that a $QBF_\forall^2$ formula ϕ is true iff the argument φ is skeptically accepted w.r.t. *pref* semantics in SF_ϕ^6 (see next lemma). The attacks between the argument $\bar{\varphi}$ and the arguments z (or $\bar{z}$ respectively) are also self-attacks, because otherwise the arguments z (or $\bar{z}$ respectively) would defend themselves against the attacks from $\bar{\varphi}$, while only φ should (actively) attack $\bar{\varphi}$. Moreover, we do not want $\bar{\varphi}$ in an admissible set, but a self-loop $(\bar{\varphi}, \bar{\varphi})$ would make SF^6_ϕ redundant. To ensure $\bar{\varphi}$ is in no admissible set we introduce φ', which would have to be attacked in order to defend $\bar{\varphi}$, but this is impossible. We have that φ' is in a preferred extension iff φ is in the extension.

Lemma 43. *Let $\phi = \forall Y \exists Z C$ be a $QBF^2_\forall$ formula with sets of propositional atoms Y and Z and a conjunctive formula φ over a set of clauses C. Then ϕ is true iff φ is skeptically accepted in SF^6_ϕ w.r.t. pref semantics.*

Proof. First note that the argument $\bar{\varphi}$ cannot be in an admissible set, as φ' is not actively attacked. As with the previous reduction we have that no argument $c \in C$ can be in an admissible set: the only active attack towards φ is from $\bar{\varphi}$, which is in no admissible set.
"$\Rightarrow$": Assume ϕ is true, i.e. for every partial interpretation $\mathcal{I}_Y \subseteq Y$ there is a partial interpretation $\mathcal{I}_Z \subseteq Z$ such that $\mathcal{I}_Y \cup \mathcal{I}_Z \models \varphi$. Note that each $\mathcal{I}_Y$ corresponds to an admissible set $S = \{y \mid y \in \mathcal{I}_Y\} \cup \{\bar{y} \mid y \in Y \setminus \mathcal{I}_Y\}$. Every admissible set $E \in adm(SF^6_\phi)$ that has $z \in E$ for some $z \in Z$ has to have $\varphi \in E$ in order to defend z against the attack from $\bar{\varphi}$. Now φ is in E iff the arguments from $Y \cup \bar{Y} \cup Z \cup \bar{Z}$ attack all arguments $c \in C$, i.e. if the corresponding interpretation $\mathcal{I} = (E \cap Y) \cup (E \cap Z)$ makes φ true. Since we assumed ϕ to be true, we know that for each partial assignment $\mathcal{I}_Y$ (and, hence, for each admissible set) there is such a partial assignment $\mathcal{I}_Z$, therefore φ is in every preferred extension of SF^6_ϕ.
"$\Leftarrow$": Assume φ is in every preferred extension of SF^6_ϕ. As we know each partial assignment $\mathcal{I}_Y \subseteq Y$ corresponds to an admissible set S in SF^6_ϕ and for each admissible set S there is an extension $E \in pref(SF^6_\phi)$ such that $E \supseteq S$, and since we know φ can only be in an admissible set E is the corresponding interpretation $\mathcal{I} = (E \cap Y) \cup (E \cap Z)$ makes φ true, we get that for each such partial assignment $\mathcal{I}_Y$ there is an assignment $\mathcal{I}_Z \subseteq Z$ such that $\mathcal{I}_Y \cup \mathcal{I}_Z \models \varphi$, i.e. ϕ is true. □

Already we have all information to pinpoint the complexity of reasoning in ϵ-symmetric SETAFs.

Theorem 18. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the complexity of the problems $Cred_\sigma$ and $Skept_\sigma$ for redundancy-free ϵ-symmetric SETAFs the complexity results in Table 4.4 hold.*

Proof. The membership for $Cred_\sigma$ for $\sigma \in \{cf, adm, naive, stb, pref, com, stg, sem\}$ follows from the general case, likewise the membership for $Skept_\sigma$ for $\sigma \in \{naive, stb, pref, stg, sem\}$ follows from the general case. The L-membership for $Cred_{grd}$ follows from Lemma 41, from the identity $Cred_{grd} = Skept_{grd} = Skept_{com}$ we get the respective membership proofs for $Skept_{grd}$ and $Skept_{com}$. The problems $Cred_\sigma$ and $Skept_\sigma$ for $\sigma \in \{stb, stg, sem\}$ already have their full hardness for symmetric AFs allowing self-attacks (see [Dvo12a, p. 85-87]). The NP-hardness of $Cred_{adm}$ follows from Lemma 42, then the hardness of $Cred_{com}$ and $Cred_{pref}$ immediately follow by the identity $Cred_{adm} = Cred_{com} = Cred_{pref}$. Finally, the Π^P_2-hardness of $Skept_{pref}$ follows from Lemma 43. □

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	in L	NP-c	in L	NP-c	NP-c	NP-c	in L	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	in L	in L	Π_2^P-c	Π_2^P-c

Table 4.4: The complexity for redundancy-free ϵ-symmetric SETAFs allowing self-attacks. The trivial results in the first line are 'yes'-instances, the trivial results in the second line are 'no'-instances.

Without self-attacks

Illustrating the restricting nature of ϵ-symmetry, the next lemma shows that naive and stable extensions coincide in self-attack-free ϵ-symmetry. This means the low complexity of reasoning w.r.t. *naive* semantics carries over to *stb* semantics.

Lemma 44. *Let $SF = (A, R)$ be a self-attack-free, ϵ-symmetric SETAF. Then we have $naive(SF) = stb(SF)$.*

Proof. We know that every stable extension is naive, it remains to show that for self-attack-free, ϵ-symmetric SETAFs every naive extension is stable. Towards contradiction assume there is a naive extension $E \in naive(SF)$ such that there is an argument $a \in A \setminus E_R^{\oplus}$. Since E is a naive extension, we have that $E \cup \{a\}$ is not conflict-free, i.e. there is an attack (T, a) such that $T \subseteq E$. But then we have that $a \in E_R^{\oplus}$, which is a contradiction. □

This suffices to establish the complexity of reasoning in self-attack-free ϵ-symmetric SETAFs for the semantics under our consideration.

Theorem 19. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ the complexity of the problems $Cred_\sigma$ and $Skept_\sigma$ for redundancy-free self-attack-free ϵ-symmetric SETAFs coincides with the complexity of irreflexive symmetric AFs, i.e. the complexity results in Table 4.5 hold.*

Proof. Since there are no self-attacks, for every argument a the set $\{a\}$ is conflict-free, which means $Cred_{cf}$ is trivially true. Moreover, since for every conflict-free set S there is a naive extension E with $E \subseteq S$ this carries over to $Cred_{naive}$. As by Lemma 44 we have that $naive(SF) = stb(SF)$, we know that also $stb(SF) = stg(SF) = sem(SF)$ for any self-attack-free ϵ-symmetric SETAF SF, since there is always at least one naive extension. Likewise we get $Cred_{stb} = Cred_{stg} = Cred_{sem}$. As every stable extension is admissible, preferred, and complete, this also carries over to $Cred_{adm}$, $Cred_{pref}$, and $Cred_{com}$.
Now by Lemma 41 we get that it suffices to check whether an argument is involved in any attack to know if it is in the grounded extension, hence, the problems $Cred_{grd}$, $Skept_{grd}$, and $Skept_{com}$ are in L.
Now note that since for every attack (T, h) the set T is conflict-free (as the SETAF is redundancy-free there cannot be an attack within T), we can construct a naive extension

E such that an arbitrary argument a that is involved in at least one attack is not in E. Hence, to decide $Skept_\sigma$ for $\sigma \in \{naive, stb, pref, stg, sem\}$ it also suffices to check whether an argument is involved in any attack, which can be done in L. □

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	trivial	trivial	trivial	trivial	trivial	in L	trivial	trivial
$Skept_\sigma$	trivial	trivial	in L	in L	in L	in L	in L	in L	in L

Table 4.5: The complexity for redundancy-free self-attack-free ϵ-symmetric SETAFs coincides with the the complexity of irreflexive symmetric AFs. The trivial results in the first line are 'yes'-instances, the trivial results in the second line are 'no'-instances.

We see that reasoning in self-attack-free ϵ-symmetric SETAFs is very efficient. Credulous reasoning is superfluous for this class, and also skeptical reasoning is not very telling, because - as established - it is closely tied to fundamental properties of the attack relation. However, we can clearly enough efficiently decide whether a redundancy-free self-attack-free SETAF is ϵ-symmetric (e.g. by just looping over the attacks and checking if for every attack there are its redirected pendants). Hence, self-attack-free ϵ-symmetry is a tractable fragment in our sense.

Theorem 20. *Self-attack-free ϵ-symmetry is a tractable fragment for all semantics under our consideration.*

4.4 Summary

We established that most of our notions of symmetry do not allow us to reason more efficiently on SETAFs. Not only is it still hard to reason in δ-symmetric SETAFs (except for a fast algorithm to reason on the grounded extension), but this even for our most restricting notion of symmetry (ϵ-symmetry), if one allows self-attacks. The fact that we cannot exploit δ-symmetry is especially surprising given the fact that it is equivalent to having a symmetric primal graph, and symmetric AFs form a tractable fragment (see Section 4.1). We established this hardness result by showing the even stronger result that every self-attack-free SETAF can be efficiently translated into a δ-symmetric SETAF, and the provided SETAF-translation is acceptance-preserving.

We identified one tractable fragment, namely self-attack-free ϵ-symmetric SETAFs. Even though technically they form such a fragment, there probably are not many ways to exploit that fact, given the very restricting nature of this definition. Credulous reasoning at the very least is superfluous on these SETAFs, as it is trivial in all semantics but grounded, and skeptical reasoning is always directly tied to simple properties of the argument in question and the attacks it is involved. The complexity results of this Chapter 4 are summarized in Table 4.6.

		adm	*stb*	*pref*	*com*	*stg*	*sem*
Self-attack-free γ- and δ-symmetric SETAFs	$Cred_\sigma$	in L	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	in L	Π_2^P-c	Π_2^P-c
γ- and δ-symmetric SETAFs	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	in L	Π_2^P-c	Π_2^P-c
Self-attack-free ϵ-symmetric SETAFs	$Cred_\sigma$	trivial	trivial	trivial	trivial	trivial	trivial
	$Skept_\sigma$	trivial	in L	in L	in L	in L	in L
ϵ-symmetric SETAFs	$Cred_\sigma$	NP-c	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	coNP-c	Π_2^P-c	P-c	Π_2^P-c	Π_2^P-c

Table 4.6: Summary of the complexity results of chapter 4. The complexity of *cf* and *naive* semantics have been omitted, as they are in L/trivial for all problems in this table.

Note that we have the full complexity for almost all problems. An exception is reasoning on the grounded extension (and for the same reason the problem $Skept_{com}$). Here we have that these problems are in L, whereas they are P-hard in the general case. This is because in our notions of symmetry an argument is in the grounded extension iff it is not involved in any attack. Moreover we have that credulous reasoning is trivial for *cf* and *naive* semantics for self-attack-free γ-, δ-, and ϵ-symmetric SETAFs, but this is already the case for self-attack-free SETAFs without any symmetry-properties.

CHAPTER 5

Bipartiteness

In this chapter we will discuss different notions of bipartiteness for SETAFs. In Section 5.1 we will recapitulate existing results for bipartite AFs, and more generally mention the connection between bipartiteness and 2-colorability. After that in Section 5.2 we will define different notions of bipartiteness or 2-colorability for SETAFs. Then, in Section 5.3 we will examine the complexity of reasoning problems in SETAFs that exhibit said properties.

Bipartite AFs and SETAFs play an important role in Argumentation Theory, as they represent the class of frameworks where two opposing parties (e.g. two agents) try to reach a consensus over a common topic. Typically, when arguing with others, the own arguments do not contradict themselves, rather they aim to counter the other's arguments. This is captured by the class of bipartite AFs.

5.1 State of the Art in AFs

We call an AF $F = (A, R)$ bipartite if the directed graph it is based on is, i.e. if it is self-attack-free and it is possible to find two conflict-free sets of arguments Y and Z such that $Y \cap Z = \emptyset$ and $Y \cup Z = A$. These two sets (Y, Z) are called a partitioning of the arguments A, Y and Z are called the partitions.

By results of [Dun07] and [Dvo12a] we know the complexity landscape of bipartite AFs for the semantics under our consideration, they are depicted in Table 5.1. Moreover in [Dun07] it was established that bipartite AFs are coherent, which follows from the well known fact that bipartite directed graphs are odd-cycle-free. We will show that these properties also hold for some of the notions we will define for SETAFs, but not for all. As can be anticipated from Table 5.1, bipartiteness forms a tractable fragment for AFs.

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	P-c	trivial	P-c	P-c	P-c	P-c	P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	P-c	P-c

Table 5.1: Complexity for bipartite (i.e. 2-colourable) AFs.

5.2 Bipartiteness in SETAFs

As in the previous chapters, we will suggest a different notion of bipartiteness that is inspired by bipartiteness for general hypergraphs. In hypergraphs, the property that is commonly denoted as 'bipartiteness' is that for a hypergraph (V, E) there is a partitioning (Y, Z) of the set of vertices V such that every edge $e \in E$ (which itself is a set of vertices) has $e \cap Y \neq \emptyset$ and $e \cap Z \neq \emptyset$ (cf. [Ber84, EH66, Sey74]). We adapt this definition for SETAFs, it generalizes the concept of 2-colorability.

Definition 38. *Let $SF = (A, R)$ be a SETAF. Then SF is α-bipartite iff there is a partitioning of A into two sets (Y, Z), such that*

- $Y \cup Z = A$,
- $Y \cap Z = \emptyset$, *and*
- *for every attack $(T, h) \in R$ we have $(T \cup \{h\}) \cap Y \neq \emptyset$ and $T \cup \{h\}) \cap Z \neq \emptyset$.*

An equivalent way to define α-bipartiteness is that there have to be two *conflict-free* sets Y and Z such that $Y \cup Z = A$ and $Y \cap Z = \emptyset$.
Again similar to the previous chapters, where we used the primal graph of a SETAF to define properties, we will define β-bipartiteness in the same manner.

Definition 39. *Let $SF = (A, R)$ be a SETAF. Then SF is β-bipartite iff its primal graph $primal(SF)$ is bipartite, i.e. iff there is a partitioning of A into two sets (Y, Z), such that*

- $Y \cup Z = A$,
- $Y \cap Z = \emptyset$, *and*
- *for every attack $(T, h) \in R$ we have either both $h \in Y$ and $T \subseteq Z$, or $h \in Z$ and $T \subseteq Y$.*

Note that both notions of bipartiteness do not allow self-loops (a, a). Moreover self-attacks (T, h) with $h \in T$ with $|T| > 1$ are not possible in β-bipartite SETAFs, whereas they can appear in α-bipartite SETAFs. We have that every β-bipartite SETAF is α-bipartite, but not vice versa. For an illustration of α-bipartite and β-bipartite SETAFs see Figure 5.1.

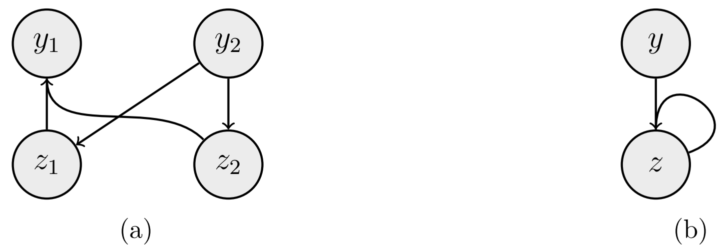


Figure 5.1: (a) A α- and β-bipartite SETAF with a partitioning $(\{y_1, y_2\}, \{z_1, z_2\})$, and (b) a α-bipartite SETAF with a partitioning $(\{y\}, \{z\})$ that is not β-bipartite.

5.3 Complexity

In the following we will find out the complexity of reasoning problems in SETAFs that exhibit our notions of bipartiteness, starting with the more general notion of α-bipartiteness, then we will investigate β-bipartiteness.

5.3.1 α-bipartiteness

We will show that we can translate every SETAF SF into an α-bipartite SETAF $Tr(SF)$ with an acceptance-preserving translation Tr. This holds for the semantics $\sigma \in \{stb, pref, com, grd, sem\}$. To this end we can reuse Tr_4 from Section 3.4.3, where we add an argument a^* as another attacker to the tail of every attack. To ensure α-bipartiteness here we add two attackers, to establish the difference between both added arguments we rename them. This way we can concatenate the translation.

Translation 6. *Let $SF = (A, R)$ be a SETAF. The SETAF translation Tr_6 is defined as $Tr_6(SF) = (A', R')$ with*

$$\begin{aligned} A' &= A \cup \{a_a^*, a_b^*\}, \\ R' &= \{(T \cup \{a_a^*, a_b^*\}, h) \mid (T, h) \in R\} \end{aligned}$$

This is the equivalent to a concatenation of two instances of Tr_4, but such that the added argument a^* is renamed to a_a^* and a_b^* respectively. As Tr_4 is efficient, clearly enough this also holds for Tr_6. It remains to show that Tr_6 is acceptance-preserving.

Lemma 45. *The SETAF-translation Tr_6 is acceptance preserving for $\sigma \Rightarrow \sigma$ with $\sigma \in \{stb, pref, com, grd, sem\}$ such that $Tr_6(SF)$ is α-bipartite for every SETAF SF.*

Proof. This follows from Lemma 34 and Lemma 35 and the fact that the arguments a_a^* and a_b^* are part of every attack, so every partitioning (Y, Z) with $a_a^* \in Y$ and $a_b^* \in Z$ shows the α-bipartiteness. □

Note that this translation does not work as a reduction for semantics based on conflict-free sets, as for every attack (T,h) in the translation the set $T \cup \{h\}$ is conflict-free. To show the hardness of our reasoning tasks for *stg* semantics we introduce another reduction from the Π_2^P-hard $QBF_\forall^2$ problem, such that the constructed SETAF is always α-bipartite. For an illustration of SF_7^Φ see Figure 5.2.

Reduction 7. *Let $\Phi = \forall Y \exists Z C$ be a $QBF_\forall^2$-formula with at least 2 clauses where in each clause at least one positive and at least one negative literal occurs, consisting of a set of clauses C over sets of propositional atoms Y and Z. We define the SETAF $SF_7^\Phi = (A,R)$, where*

$$\begin{aligned}
A &= \{\varphi, \bar{\varphi}', \bar{\varphi}, \varphi', \varphi'', \varphi'''\} \cup C \cup Y \cup \bar{Y} \cup Z \cup \bar{Z} \cup \{y', y'', y''', \bar{y}', \bar{y}'', \bar{y}''' \mid y \in Y\},\\
R &= \{(x,\bar{x}), (\bar{x},x) \mid x \in Y \cup Z\} \cup \{(\{x \mid \bar{x} \in c\} \cup \{\bar{x} \mid x \in c\}, c) \mid c \in C\} \cup\\
&\quad \{(\{c \mid c \in C\}, \bar{\varphi}'), (\bar{\varphi}', \varphi), (\bar{\varphi}, \varphi), (\varphi, \bar{\varphi})\} \cup\\
&\quad \{(\{\varphi, \varphi'\}, \varphi''), (\{\varphi, \varphi'\}, \varphi'''), (\{\varphi'', \varphi'''\}, \varphi''), (\{\varphi'', \varphi'''\}, \varphi''')\} \cup\\
&\quad \{(\{y, y'\}, y''), (\{y, y'\}, y'''), (\{y'', y'''\}, y''), (\{y'', y'''\}, y''') \mid y \in Y\} \cup\\
&\quad \{(\{\bar{y}, \bar{y}'\}, \bar{y}''), (\{\bar{y}, \bar{y}'\}, \bar{y}'''), (\{\bar{y}'', \bar{y}'''\}, \bar{y}''), (\{\bar{y}'', \bar{y}'''\}, \bar{y}''') \mid \bar{y} \in \bar{Y}\}
\end{aligned}$$

We have (as we will show in Lemma 47) that arguments y' and $\bar{y}'$ are in every *stg* extension, and the arguments y'' and y''' (or $\bar{y}''$ and $\bar{y}'''$ respectively) cannot be in a conflict-free set together, so the only way to have both in the range of a *stg* extension is to have y (or $\bar{y}$ respectively) in this extension. This way every combination of arguments from Y and $\bar{Y}$ (that correspond to a partial interpretation over variables Y) is in an incomparable *stg* extension.
It is not immediate why SF_7^Φ is always α-bipartite; for this we need to have for each clause $c \in C$ to have at least one positive and at least one negative literal, as otherwise this partitioning could produce a monochromatic edge (i.e. an edge such that all involved arguments are in just one of Y or Z). Moreover we assume there are at least two clauses; these two constraints do not affect the hardness of the $QBF_\forall^2$ problem. Consider a partitioning (A,B) where $A = (\{c_x, \bar{\varphi}, \bar{\varphi}', \varphi', \varphi''\} \cup \{\bar{y}, y', y'', \bar{y}'' \mid y \in Y\} \cup \{\bar{z} \mid z \in Z\})$ and $B = (\{c \mid c \in C \setminus \{c_x\}\} \cup \{\varphi, \varphi'''\} \cup \{y, \bar{y}', \bar{y}'', y'' \mid y \in Y\} \cup \{z \mid z \in Z\} \cup \{\varphi'''\})$, where c_x is an arbitrary clause. Then one can check that (A,B) is a partitioning such that SF_7^Φ is α-bipartite (the coloring in Figure 5.2 corresponds to such a partitioning). The following proof follows the structure of [Dvo12a, p. 52-55].

Lemma 46. *Let Φ be a $QBF_\forall^2$ formula and let $SF_7^\Phi = (A,R)$, then for every extension $E \in stg(SF_7^\Phi)$ we have $\{\varphi'', \varphi'''\} \not\subseteq E$, $\{y'', y'''\} \not\subseteq E$, and $\{\bar{y}'', \bar{y}'''\} \not\subseteq E$ for each $y \in Y$. Moreover we have $x \in E$ iff $\bar{x} \notin E$ for each $x \in Y \cup Z \cup \{\varphi\}$.*

Proof. The first statement immediately follows from the fact that E is conflict-free. Moreover we have that at at least one of x and $\bar{x}$ is in E: towards contradiction assume otherwise, i.e. $\{x,\bar{x}\} \cap E = \emptyset$. If $x = \varphi$, then $E' = E \cup \{\bar{\varphi}\}$ is conflict-free with $E_R'^\oplus \supset E_R^\oplus$. If $x \in Y \cup Z$, then $E' = (E \setminus \{c \mid c \in C, \text{there is some } (T,c) \in R \text{ such that } T \subseteq$

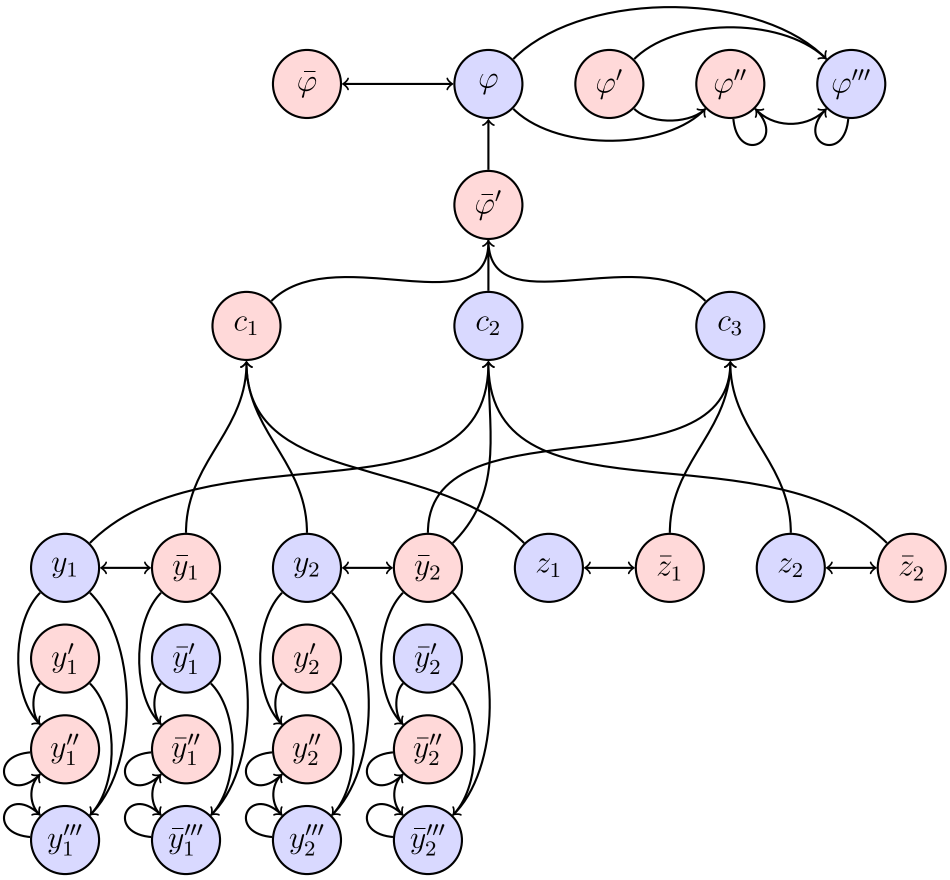


Figure 5.2: Illustration of SF_7^Φ for $\Phi = \forall Y \exists Z \varphi(Y, Z)$ with $Y = \{y_1, y_2\}$, $Z = \{z_1, z_2\}$, and $\varphi = \{\{y_1, \bar{y}_2, \bar{z}_1\}, \{\bar{y}_1, y_2, z_2)\}, \{y_2, z_1, \bar{z}_2\}\}$. The coloring of the arguments corresponds to a possible partitioning that shows the α-bipartiteness of SF_7^Φ, i.e. we have that no attack is monochromatic.

$E \cup \{x\}\} \cup \{x\}$ is conflict-free with $E_R'^{\oplus} \supset E_R^{\oplus}$. By conflict-freeness we also have that at most one of x and $\bar{x}$ is in E. □

Lemma 47. *Let Φ be a $QBF_\forall^2$ formula and let $SF_7^\Phi = (A, R)$, then $\{x' \mid x \in Y \cup \bar{Y} \cup \{\varphi\}\} \subseteq E$ for every $E \in stg(SF_7^\Phi)$.*

Proof. Towards contradiction assume $E \in stg(SF_7^\Phi)$ and $x' \notin E$ for some $x \in Y \cup \bar{Y} \cup \{\varphi\}$, then we have $E' = (E \cup \{x'\}) \setminus \{x'', x'''\} \in cf(SF_7^\Phi)$ with $E_R'^{\oplus} \supset E_R^{\oplus}$, which is a contradiction to the assumption $E \in stg(SF_7^\Phi)$. □

Lemma 48. *Let Φ be a $QBF_\forall^2$ formula and let $SF_7^\Phi = (A, R)$, then φ is in every stg extension iff Φ is true.*

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	NP-c	trivial	NP-c	NP-c	NP-c	P-c	Σ_2^P-c	Σ_2^P-c
$Skept_\sigma$	trivial	trivial	in L	coNP-c	Π_2^P-c	P-c	P-c	Π_2^P-c	Π_2^P-c

Table 5.2: The complexity of α-bipartite SETAFs.

Proof. "⇒": Assume Φ is false, we show that then there is an extension $E \in stg(SF_7^\Phi)$ such that $\varphi \notin E$. As Φ is false, there is a partial interpretation I_Y such that for each partial interpretation I_Z we have that at least one clause is not true, i.e. in the corresponding set of arguments at least one argument $c \in C$ is attacked. As by Lemma 46 and since $\bar{\varphi}'$ is not attacked, the only way to have $\{y'', y''' \mid y \in I_Y\} \cup \{\bar{\varphi}'\} \subseteq E_R^\oplus$ is if we also have $\bar{\varphi}' \in E$, we know that such a stage extension E with $\bar{\varphi}' \in E$ exists, but this extension can only have $\varphi \notin E$.
"⇐": Assume Φ is true, and let, towards contradiction, $E \in stg(SF_7^\Phi)$ with $\varphi \notin E$. We know that for each partial interpretation I_Y there is a partial interpretation I_Z such that $I_Y \cup I_Z$ makes φ true. Let $I_Y = E \cap Y$ and let I_Z be such a partial interpretation such that $I_Y \cup I_Z$ makes φ true. Moreover let $E' = I_Y \cup I_Z \cup \{\bar{x} \mid x \in (Y \cup Z) \setminus (I_Y \cup I_Z)\} \cup C \cup (E \cap (Y' \cup Y'' \cup Y''' \cup \bar{Y}' \cup \bar{Y}'' \cup \bar{Y}''')) \cup \{\varphi, \varphi'\}$. One can check that E' is conflict-free in SF_7^Φ, also we have $E_R'^\oplus \supset E_R^\oplus$: by construction the ranges of E' and E coincide on all arguments but arguments $c \in C$ and on the arguments φ'' and φ''', where we have $C \subseteq E_R'^\oplus$ and $\{\varphi'', \varphi'''\} \subseteq E_R'^\oplus$, but $\{\varphi'', \varphi'''\} \not\subseteq E_R^\oplus$. This is a contradiction to the assumption $E \in stg(SF_7^\Phi)$. □

These results give us the complexity landscape for α-bipartite SETAFs; they have the full complexity, i.e. α-bipartiteness does not allow us to reason more efficiently.

Theorem 21. *For $\sigma \in \{cf, adm, grd, naive, stb, com, pref, stg, sem\}$ for the problems $Cred_\sigma$ and $Skept_\sigma$ for α-bipartite SETAFs the complexity results in Table 5.2 hold.*

Proof. The membership follows from the general case, the trivial results for $Cred_{cf}$ and $Cred_{naive}$ follow from the fact that α-bipartite SETAFs cannot contain self-loops. We obtain the hardness for $Cred_\sigma$ and for $Skept_\sigma$ with $\sigma \in \{stb, pref, com, grd, sem\}$ by Lemma 45. The hardness of $Cred_{adm}$ follows from the identity $Cred_{adm} = Cred_{pref}$. The hardness of $Cred_{stg}$ follows from Lemma 48, likewise the hardness of $Skept_{stg}$ follows from Lemma 48 and the fact that by Lemma 46 we then have $\bar{\varphi}$ is in every extension $E \in stg(SF_7^\Phi)$ iff Φ is false. □

5.3.2 β-bipartiteness

For bipartite AFs we know that *pref* semantics coincides with *stb* [Dun07] (and therefore with *sem* and *stg*), as they are odd-cycle-free and therefore by Lemma 7 they are limited controversial and coherent (see [NP06, p. 66]), this also applies for SETAFs. Moreover credulous and skeptical reasoning for these semantics (and therefore for all semantics

Algorithm 5.1: Compute the set of credulously accepted arguments w.r.t. *pref* semantics

Input : A β-bipartite SETAF $SF = (A, R)$ with a partitioning (Y, Z)
Output : The admissible set Y_i of credulously accepted arguments

1 $i := 0$
2 $Y_0 := Y$
3 $R_0 := R$
4 **repeat**
5 $\quad i := i + 1$
6 $\quad Y_i := Y_{i-1} \setminus \{y \mid y \in Y_{i-1},$ there is some $(Z', y) \in R_{i-1}$ with $Z' \subseteq Z$ such that $\forall z \in Z' \ |\{(Y', z) \mid (Y', z) \in R_{i-1}\}| = 0\}$
7 $\quad R_i := R_{i-1} \setminus \{(Y', z) \mid Y' \subseteq Y, z \in Z, Y' \not\subseteq Y_i\}$
8 **until** $Y_i = Y_{i-1}$;

under our consideration) are tractable in AFs. We are going to adapt the algorithm that computes the set of credulously accepted arguments w.r.t. *pref* semantics from [Dun07, p. 707] for SETAFs, and moreover we will show that these properties also hold for β-bipartite SETAFs.

As in the algorithm for AFs, our adaptation iteratively removes arguments that cannot be defended. This algorithm has to be executed for both the set Y and the set Z to get all credulously accepted arguments of a SETAF SF. Assume we start with Y. In step 6 of the i-th iteration of the of the algorithm we remove every argument y that is attacked via an attack (Z', y) (as SF is β-bipartite Z' must be a subset of Z) such that there are no defenders against the attack left, i.e. no $z \in Z'$ is attacked by a subset of the arguments left in Y_{i-1}. In step 7 we remove all attacks that origin from already removed arguments; they cannot be part of a defending attack.
An example run for Algorithm 5.1 on Example 4a and Example 4b can be found in Table 5.3, for an illustration of the respective SETAFs see Figure 5.3.

Example 4a. *Consider the SETAF* $SF_1 = (A, R)$*, with* $A = \{y_1, y_2, y_3, y_4, z_1, z_2, z_3, z_4\}$ *and* $R = \{(\{z_1, z_2\}, y_1), (y_1, z_3), (z_2, y_2), (\{z_2, z_3\}, y_3), (y_3, z_4), (y_4, z_4), (z_4, y_4)\}$.

Example 4b. *Consider the SETAF* $SF_2 = (A, R)$*, with* $A = \{y_1, y_2, y_3, y_4, z_1, z_2, z_3, z_4\}$ *and* $R = \{(\{z_1, z_2\}, y_1), (y_1, z_3), (z_2, y_2), (\{z_2, z_3, y_4\}, y_3), (y_3, z_4), (y_4, z_4), (z_4, y_4)\}$.

We have that $(\{y_1, y_2, y_3, y_4\}, \{z_1, z_2, z_3, z_4\})$ is a partitioning that shows the SETAF from Example 4a is β-bipartite, and the SETAF from Example 4b is α-bipartite. The latter cannot be β-bipartite with any partitioning, as it contains an odd cycle (y_3, z_4, y_4). In the first example run of Algorithm 5.1 in Table 5.3 we see that the algorithm correctly identifies those arguments $Y' \subseteq Y$ that are credulously accepted. In the second example we try to run the algorithm on an α-bipartite SETAF that is not β-bipartite. At iteration $i = 2$ we see that no argument $y \in Y$ is removed any more, which would suggest that

Example run of Algorithm 5.1 on $Y = \{y_1, y_2, y_3, y_4\}$ from SETAF SF_1 of Example 4a:

i	Y_i		R_i
0	Y	$= \{y_1, y_2, y_3, y_4\}$	R
1	$Y_0 \setminus \{y_1, y_2\}$	$= \{y_3, y_4\}$	$R_0 \setminus \{(y_1, z_3)\}$
2	$Y_1 \setminus \{y_3\}$	$= \{y_4\}$	$R_1 \setminus \{(y_3, z_4)\}$
3	Y_2	$= \{y_4\}$	R_2

Example run of Algorithm 5.1 on $Y = \{y_1, y_2, y_3, y_4\}$ from SETAF SF_2 of Example 4b:

i	Y_i		R_i
0	Y	$= \{y_1, y_2, y_3, y_4\}$	R
1	$Y_0 \setminus \{y_1, y_2\}$	$= \{y_3, y_4\}$	$R_0 \setminus \{(y_1, z_3)\}$
2	Y_1	$= \{y_3, y_4\}$	R_1

Table 5.3: Two example runs of Algorithm 5.1. The second run is not on a β-bipartite SETAF and does not return only credulously accepted arguments.

$Y_2 = \{y_3, y_4\}$ are those arguments from Y that are credulously accepted.
The argument y_3 is not removed like in the first run of the algorithm, because there is still the attack $(\{z_2, z_3, y_4\}, y_3)$ towards y_3 that is not 'flagged' as inapplicable for defending attacks. However, the respective SETAF SF_2 from Example 4b has exactly one preferred extension that is $\{y_4, z_1, z_2, z_3\}$, i.e. y_3 is *not* credulously accepted, as the algorithm suggests, which shows that the algorithm is not applicable for SETAFs that are not β-bipartite.
In what follows we will show that it is indeed correct for β-bipartite SETAFs.

Lemma 49 (cf.[Dun07, p. 707-708])**.** *Let $SF = (A, R)$ be a β-bipartite SETAF with a partitioning (Y, Z), then an argument $a \in Y$ is credulously accepted w.r.t. pref semantics iff it is in the set returned by Algorithm 5.1.*
Moreover the set returned by Algorithm 5.1 is admissible in SF.

Proof. "$\Rightarrow$": We will show inductively that for every iteration of the algorithm the arguments that are removed in step 6 are not defensible and the attacks that are removed in step 7 cannot be part of a defending attack. For the first iteration this is the case,

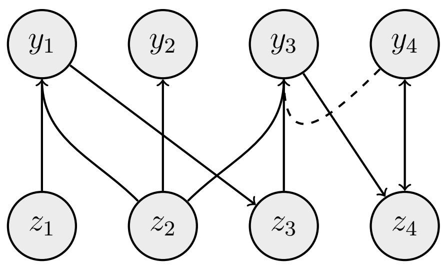


Figure 5.3: The SETAF from Example 4a (without the dashed line), and the SETAF from Example 4b (with the dashed line).

as we construct Y_1 by only removing those arguments $y \in Y$ from Y that are attacked by an attack (Z', y) on which no counter-attack exists. Moreover we remove all attacks (Y', z) towards arguments $z \in Z$ such that for one of the arguments $y' \in Y'$ we already showed it is not defensible, as they cannot defend any argument in an admissible set. Likewise, assuming this property holds for the $i-1$-th iteration, in the i-th iteration we only remove arguments that are not defensible and attacks that cannot play a role in admissible sets.
Assume towards contradiction an argument $y \in Y$ is credulously accepted, but not in the set S that is returned by the algorithm. This means at some iteration i the argument y is removed, but, as established, this means it is not defensible, which is a contradiction to the assumption is it credulously accepted.
"$\Leftarrow$": Let S be the set that is returned by the algorithm. Assume we have $x \in S$ for some argument $x \in Y$. As we have $S \subseteq Y$, we know S is conflict-free in SF. Moreover we know that S defends x: towards contradiction assume otherwise, i.e. there is an attack (Z', x) towards x such that S does not attack Z'. But then x would be removed in step 6, which is a contradiction to the assumption that $x \in S$. □

Algorithm 5.1 runs in polynomial time: there can be at most $|Y|$ iterations; step 6 is efficient, as all involved sets are bounded by the number of attacks and the number of arguments involved in an attack; step 7 is also efficient, as it suffices to check for every attack towards arguments $z \in Z$.
Note that by symmetry this algorithm also works for arguments $z \in Z$ such that it is sufficient to compute all credulously accepted arguments of a β-bipartite SETAF SF. Hence, $Cred_{pref}$ is P-easy for this subclass. We will now show that this result carries over to other semantics under our consideration.

Lemma 50. *Let $SF = (A, R)$ be a β-bipartite SETAF with a partitioning (Y, Z), then we have $pref(SF) = stb(SF)$.*

Proof. As we know every stable extension is preferred, it suffices to show that for β-bipartite SETAFs also every preferred extension is stable. We know that β-bipartite SETAFs have no odd-cycles, so by Lemma 7 they are limited controversial and therefore coherent (see [NP06, p. 66]), which implies $pref(SF) = stb(SF)$. □

Note that as there always is at least one preferred extension there also always is a stable extension, which further implies $stb(SF) = sem(SF) = stg(SF)$. The following lemma also holds for AFs (see [VP00]).

Lemma 51. *Let $SF = (A, R)$ be a SETAF with $pref(SF) = stb(SF)$. Then an argument $a \in A$ is skeptically accepted w.r.t. pref semantics iff for every attack $(T, a) \in R$ towards a we have that $T \not\subseteq E$ for every preferred extension $E \in pref(SF)$.*

Proof. "$\Rightarrow$": Assume an argument $a \in A$ is in every preferred extension of SF and let (T, a) be an arbitrary attack towards a. Then T cannot be a subset of any preferred

extension E, as we would have $T \cup \{a\} \subseteq E$, which is not conflict-free.
"$\Leftarrow$": Assume in every extension $E \in \mathit{pref}(SF)$ we have for every attack (T, a) towards a that $T \nsubseteq E$. This means for every attack there is some $t \in T$ such that $t \notin E$. But as E is stable by assumption, this means t is attacked, and, hence, a is defended against all attacks. □

By a result of [Dun07, p. 708] we know that even for bipartite AFs $F = (A, R)$ with a partitioning (Y, Z) it is NP-complete to decide for sets $S \subseteq A$ if the arguments are jointly credulously accepted w.r.t. *pref* semantics. This hardness-result carries over to SETAFs. However, if we restrict the problem to deciding whether a set $S \subseteq Y$ is jointly credulously accepted, this problem becomes P-easy even for SETAFs, as this is the case iff every single argument $a \in S$ is credulously accepted, which we established can be decided in polynomial time with Algorithm 5.1. This membership-proof also applies to AFs.

Lemma 52. *Let $SF = (A, R)$ be a β-bipartite SETAF with a partitioning (Y, Z). Then for any set $Y' \subseteq Y$ there is a preferred extension $E \supseteq Y'$ iff every argument $y' \in Y'$ is credulously accepted w.r.t. pref semantics.*

Proof. By Lemma 49 we have that an argument $a \in Y$ is credulously accepted w.r.t. *pref* semantics iff it is in the set S returned by Algorithm 5.1, and we have $S \in \mathit{adm}(SF)$. This means that all credulously accepted arguments $a \in Y$ are also jointly credulously accepted in SF, which also means that every subset $Y' \subseteq Y$ that consists only of credulously accepted arguments is jointly credulously accepted w.r.t. *adm*, which in turn means they are jointly credulously accepted w.r.t. *pref* semantics, as every admissible set is part of a subset-maximal admissible set. □

Again, by symmetry, this result also applies for sets $Z' \subseteq Z$.

Theorem 22. *For $\sigma \in \{\mathit{cf}, \mathit{adm}, \mathit{grd}, \mathit{naive}, \mathit{stb}, \mathit{com}, \mathit{pref}, \mathit{stg}, \mathit{sem}\}$ the complexity of the problems Cred_σ and Skept_σ for β-bipartite SETAFs the complexity results in Table 5.4 hold.*

Proof. The respective hardness proofs follow from the hardness of bipartite AFs. As already established, Algorithm 5.1 can be used to efficiently (namely, in polynomial time) compute the set of credulously accepted arguments w.r.t. *pref* semantics, which carries over to *com* and *adm*, and since by Lemma 50 we know that preferred and stable semantics coincide, also to *stb*, *stg*, and *sem*. For the P-membership of Skept_σ for $\sigma \in \{\mathit{stb}, \mathit{pref}, \mathit{stg}, \mathit{sem}\}$ we use the same identity $\mathit{stb}(SF) = \mathit{pref}(SF) = \mathit{stg}(SF) = \mathit{sem}(SF)$, and note that to check if an argument $a \in A$ is skeptically accepted w.r.t. *pref* semantics by Lemma 51 we know that it suffices to check if for every attack (T, a) towards a the set T is jointly credulously accepted, which, by Lemma 52, can be done in polynomial time, as it suffices to check if every argument in T is credulously accepted. Finally, the L membership of $\mathit{Skept}_{\mathit{naive}}$ follows from the general case, and the trivial results for $\mathit{Cred}_{\mathit{cf}}$ and $\mathit{Cred}_{\mathit{naive}}$ follow from the fact that every β-bipartite SETAF has

	cf	*adm*	*naive*	*stb*	*pref*	*com*	*grd*	*stg*	*sem*
$Cred_\sigma$	trivial	P-c	trivial	P-c	P-c	P-c	P-c	P-c	P-c
$Skept_\sigma$	trivial	trivial	in L	P-c	P-c	P-c	P-c	P-c	P-c

Table 5.4: The complexity of β-bipartite SETAFs coincides with the the complexity of bipartite AFs. The trivial results in the first line are 'yes'-instances, the trivial results in the second line are 'no'-instances.

no self-loops (i.e. every argument $a \in A$ is in the conflict-free set $\{a\}$, and, hence, in a naive extension). □

We showed that reasoning in β-bipartite SETAFs is tractable. It remains to show that β-bipartiteness is also a property that we can detect efficiently, and moreover that we can efficiently compute such a partitioning (Y, Z), as it is required to be fixed already before applying the discussed algorithms. As we established that a SETAF is β-bipartite iff its primal graph is bipartite, we know that this is efficiently computable: we can check in polynomial time if a directed graph has self-loops, and since every self-loop-free directed graph is bipartite iff the corresponding undirected graph is, we can use depth-first-search to assign the arguments to the sets Y and Z or to detect an odd cycle in linear time. This leads us to the following result.

Theorem 23. *β-bipartiteness is a tractable fragment for all semantics under our consideration.*

5.4 Summary

In this chapter we defined two notions of bipartiteness/2-colorability for SETAFs. We established that α-bipartiteness has full complexity, except for the problems $Cred_{cf}$ and $Cred_{naive}$. As α-bipartite SETAFs are self-loop-free, every argument is in a conflict-free set and therefore in a naive extension, making these problems trivially true.

Furthermore we showed that β-bipartiteness indeed allows us to reason more efficiently, namely the complexity results for β-bipartite SETAFs coincide with the results of bipar-

		adm	*naive*	*stb*	*pref*	*com*	*stg*	*sem*
α-bipartite SETAFs	$Cred_\sigma$	NP-c	trivial	NP-c	NP-c	NP-c	Σ_2^P-c	Σ_2^P-c
	$Skept_\sigma$	trivial	in L	coNP-c	Π_2^P-c	P-c	Π_2^P-c	Π_2^P-c
β-bipartite SETAFs	$Cred_\sigma$	P-c	trivial	P-c	P-c	P-c	P-c	P-c
	$Skept_\sigma$	trivial	in L	P-c	P-c	P-c	P-c	P-c

Table 5.5: Summary of the complexity results of chapter 5. The complexity of *cf* semantics has been omitted, as all problems are trivial, and the complexity of *grd* has been omitted, as it is P-complete for all problems in this table.

tite AFs. Moreover we established that some properties such as coherency carry over from bipartite AFs (c.f. [Dun07]) to β-bipartite SETAFs. This means we identified a tractable fragment for SETAFs. For all complexity results of this chapter see Table 5.5.

Beside that we established that deciding for a set of arguments if it is in an admissible set for β-bipartite SETAFs (or in a bipartite AFs) is efficiently decidable, if all arguments are in the same partition (it was shown in [Dun07] that this is NP-hard even for AFs if the arguments are in different partitions).

CHAPTER 6

Conclusion

In this final chapter we will summarize our findings, draw conclusions, and set the results of this thesis in the respective contexts. Moreover we will give an overview of related elaborations and an outlook to open questions that could be examined in the future.

6.1 Summary

The contributions of this thesis are twofold: first we introduced graph-properties for SETAFs that generalize known properties of AFs, then we investigated the complexity of reasoning tasks in the resulting classes of frameworks.

6.1.1 Graph-classes and their Properties

First we adapted useful concepts and notions such as translations and their properties for argumentation frameworks with collective attacks. In the later chapters we utilized these generalizations when we investigated the complexity of reasoning problems on SETAFs.

We defined several degrees of acyclicity for directed hypergraphs. The starting point for some of these notions was the hypergraph literature; additionally we introduced strong-attack-cycle-freeness, motivated by the nature of consecutive attacks. Moreover we argued why some notions are not suitable for SETAFs. Furthermore we introduced the maximal and minimal cycle length as properties of a SETAF, and adapted the notion of even-cycle-freeness and odd-cycle-freeness for SETAFs.

Based on properties of symmetric AFs we generalized different notions of symmetry for SETAFs, and identified unfruitful approaches. Then we showed properties of SETAFs that belong to the respective sub-classes, in particular we established that an argument is in the grounded extension only if it is not involved in any attack (for all of our notions of symmetry); this property carries over to AFs.

For bipartiteness/2-colorability we gave two different definitions for SETAFs, and investigated fundamental properties of frameworks of the resulting classes. Moreover we showed generically how certain SETAFs can be partitioned (i.e. "coloured") such that at most two partitions (colours) are needed.

6.1.2 Complexity Analysis

Having those new notions at hand we examined the complexity of credulous and skeptical reasoning thoroughly, identifying several tractable fragments and establishing which notions do not allow us to reason more efficiently. The identified tractable fragments are

- acyclicity,
- a maximal cycle length of 1 (for all semantics under our consideration but *stg*),
- even-cycle-freeness,
- self-attack-free ϵ-symmetry, and
- β-bipartiteness.

For tractable fragments we do not only require the reasoning tasks to be efficiently decidable, but also the properties themselves - strictly speaking to decide whether a SETAF belongs to a class that allows us efficient reasoning - have to be efficiently decidable. For the listed classes this was established as well, concluding the argument that these are indeed tractable fragments.

Our starting point when defining graph properties for SETAFs was often the primal graph. We called a SETAF acyclic if its primal graph is acyclic, δ-symmetric if its primal graph is symmetric, and β-bipartite if its primal graph is bipartite. For acyclicity and bipartiteness this natural notion is a tractable fragment for both AFs and SETAFs. Maybe surprisingly this is not the case for symmetry: here we established full complexity for our reasoning problems, even if the framework in question enjoys δ-symmetry.

At the same time we introduced practical translations. We mainly used them to establish complexity results, but they also give insights about the expressive character of certain restrictions. We gave translations that allow us to transform every SETAF into a framework with no short cycles, that has no strong-attack-cycles or γ'-cycles, enjoys δ-symmetry, or can be partitioned in the sense of α-bipartiteness. The fact that this is possible for every SETAF tells us more about these sub-classes and the expressiveness of the proposed restrictions.

Beside translations, to establish our hardness results we sometimes gave explicit reductions. In particular we introduced a new "standard reduction" (Reduction 3) for SETAFs that exploits the expressive power of collective attacks and corresponds to the standard reduction in AFs. We adapted it to prove hardness for skeptical acceptance in the preferred semantics for SETAFs with no cycles that consist of more than 4 arguments (Reduction 4), and to show the hardness of reasoning w.r.t. stage semantics in α-bipartite SETAFs (Reduction 7).

To perform complexity analysis, it is often useful to define reductions starting from restricted instances and to exploit certain existing structures in them. Throughout our investigations we identified various problems that allow this. To this end we discovered hard subclasses for different notions of acyclicity, symmetry, and bipartiteness.

6.2 Related Work

In general there have been many efforts to investigate the computational aspects of AFs [DD17, Dvo12a, Dun07] and SETAFs [DGW18]. Tractable fragments have been investigated for AFs [Dvo12a, Dun07, DB01, CMDM05, DD17] and extensions thereof such as Abstract Dialectical Frameworks (ADFs) [DKZLW20, BW10]. Other approaches to achieve a computational speed-up have been made: the complexity of reasoning has been studied by means of parametrized complexity [Dvo12a, DPW12, Dvo12b], allowing efficient computations for frameworks with fixed parameters.

Complexity analysis for Argumentation Frameworks has been performed for various semantics and reasoning modes that have not been discussed in this thesis. To this end see e.g. ideal [Dun09, Dvo12a] or eager reasoning [Cam07], *cf2* or *stage2* semantics [DG12, GW13], or resolution-based grounded semantics [BDG09, Dvo12a].

Concerning properties of directed hypergraphs the reader is referred to [GLP93], for an overview of hypergraphs in general see [Ber73]. Graph-properties have been studied for general (i.e. not directed) hypergraphs [Ber84, Sey74], to a large extend this was done in the context of database theory [Fag83].

6.3 Future Work

The search for tractable fragments in Argumentation is essential, as most reasoning tasks are hard for the first or second level of the polynomial hierarchy. Naturally, this is an invitation to examine the complexity of restricted sub-classes not only for SETAFs, but for all kinds of extensions of Argumentation Frameworks, such as Claim-Augmented Argumentation Frameworks (CAFs) [DW20], Argumentation framework with recursive attacks (AFRAs) [BCGG11], Extended Argumentation Frameworks (EAFs) [DMB10], or others.

We know it is possible to translate a SETAF into an AF with similar semantic properties (cf. [Pol17, FB19]). However, it is not known which graph-properties are preserved during such a translation. This is an alternative approach to give an upper bound on the complexity of reasoning in these graph-classes, as well as it gives us insights about the graph-properties and SETAFs in general.

The results from this thesis can be extended to other decision problems over SETAFs such as verification of extensions, set acceptance, coherence-checking, existence of (non-empty) extensions, or labelling based reasoning (cf. [BCG11]). Moreover additional semantics can be considered w.r.t. the graph-classes that were investigated in this thesis.

Furthermore reasoning in graph-classes can be investigated in terms of parametrized complexity and backdoor approaches (for similar investigations in AFs see [OS11]) that measure the distance to tractable classes.

Finally, we will state some concrete examples for open problems. In particular, we know that reasoning is still hard for k-partite AFs for certain admissible-based semantics [Dun07], this can be investigated for other semantics such as stage. Moreover we do not know for AFs with a maximal cycle length of 4 whether the problem $Skept_{pref}$ is Π_2^P-hard or in coNP (for SETAFs we showed Π_2^P-hardness).

List of Figures

List of Tables

Bibliography

[BCG11] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *Knowledge Eng. Review*, 26(4):365–410, 2011.

[BCGG11] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. Afra: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52(1):19 – 37, 2011.

[BD07] Trevor J. M. Bench-Capon and Paul E. Dunne. Argumentation in artificial intelligence. *Artif. Intell.*, 171(10-15):619–641, 2007.

[BDG09] Pietro Baroni, Paul E. Dunne, and Massimiliano Giacomin. Computational properties of resolution-based grounded semantics. In Craig Boutilier, editor, *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 683–689, 2009.

[BDKT97] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.

[Ber73] Claude Berge. *Graphs and Hypergraphs*. North-Holland, 1973.

[Ber84] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[BPS09] Trevor J. M. Bench-Capon, Henry Prakken, and Giovanni Sartor. Argumentation in legal reasoning. In Guillermo Ricardo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 363–382. Springer, 2009.

[BW10] Gerhard Brewka and Stefan Woltran. Abstract dialectical frameworks. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczynski, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*. AAAI Press, 2010.

[Cam07] Martin Caminada. Comparing two unique extension semantics for formal argumentation: Ideal and eager. *Proc. of the 19th Belgian-Dutch Conference on Artificial Intelligence*, 2007.

[CMDM05] Sylvie Coste-Marquis, Caroline Devred, and Pierre Marquis. Symmetric argumentation frameworks. In Lluís Godo, editor, *Proceedings of Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 317–328, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[DB01] Paul E. Dunne and Trevor J.M. Bench-Capon. Complexity and combinatorial properties of argument systems. Technical report, Dept. of Computer Science, University of Liverpool, 2001.

[DD17] Wolfgang Dvořák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. *FLAP*, 4(8), 2017.

[DG12] Wolfgang Dvořák and Sarah Alice Gaggl. Computational aspects of cf2 and stage2 argumentation semantics. In Bart Verheij, Stefan Szeider, and Stefan Woltran, editors, *Computational Models of Argument - Proceedings of COMMA 2012, Vienna, Austria, September 10-12, 2012*, volume 245 of *Frontiers in Artificial Intelligence and Applications*, pages 273–284. IOS Press, 2012.

[DGW18] Wolfgang Dvořák, Alexander Greßler, and Stefan Woltran. Evaluating SETAFs via answer-set programming. In Matthias Thimm, Federico Cerutti, and Mauro Vallati, editors, *Proceedings of the Second International Workshop on Systems and Algorithms for Formal Argumentation (SAFA 2018) co-located with the 7th International Conference on Computational Models of Argument (COMMA 2018), Warsaw, Poland, September 11, 2018*, volume 2171 of *CEUR Workshop Proceedings*, pages 10–21. CEUR-WS.org, 2018.

[DKZLW20] Martin Diller, Atefeh Keshavarzi Zafarghandi, Thomas Linsbichler, and Stefan Woltran. Investigating subclasses of abstract dialectical frameworks. *Argument & Computation*, 11:191–219, 2020.

[DMB10] Paul E. Dunne, Sanjay Modgil, and Trevor J. M. Bench-Capon. Computation in extended argumentation frameworks. In Helder Coelho, Rudi Studer, and Michael J. Wooldridge, editors, *ECAI 2010 - 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 2010, Proceedings*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 119–124. IOS Press, 2010.

[DPW12] Wolfgang Dvořák, Reinhard Pichler, and Stefan Woltran. Towards fixed-parameter tractable algorithms for abstract argumentation. *Artif. Intell.*, 186:1–37, 2012.

[DRW20] Wolfgang Dvořák, Anna Rapberger, and Stefan Woltran. On the different types of collective attacks in abstract argumentation: Equivalence results for SETAFs. *Journal of Logic and Computation*, 30(5):1063–1107, 2020.

[Dun95] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[Dun07] Paul E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artif. Intell.*, 171(10-15):701–729, 2007.

[Dun09] Paul E. Dunne. The computational complexity of ideal semantics. *Artif. Intell.*, 173(18):1559–1591, 2009.

[Dvo12a] Wolfgang Dvořák. *Computational Aspects of Abstract Argumentation*. PhD thesis, Vienna University of Technology, Institute of Information Systems, 2012.

[Dvo12b] Wolfgang Dvořák. Technical note: Exploring Σ_2^P / Π_2^P-hardness for argumentation problems with fixed distance to tractable classes. *CoRR*, abs/1201.0478, 2012.

[DW11] Wolfgang Dvořák and Stefan Woltran. On the intertranslatability of argumentation semantics. *J. Artif. Intell. Res.*, 41:445–475, 2011.

[DW20] Wolfgang Dvořák and Stefan Woltran. Complexity of abstract argumentation under a claim-centric view. *Artif. Intell.*, 285:103290, 2020.

[EG93] Thomas Eiter and Georg Gottlob. Propositional circumscription and extended closed-world reasoning are Π_2^P-complete. *Theoretical Computer Science*, 114(2):231–245, 1993.

[EH66] Paul Erdős and András Hajnal. On chromatic number of graphs and set-systems. *Acta Mathematica Academiae Scientiarum Hungarica*, 17(1):61–99, 1966.

[Fag83] Ronald Fagin. Degrees of acyclicity for hypergraphs and relational database schemes. *J. ACM*, 30:514–550, 1983.

[FB19] Giorgos Flouris and Antonis Bikakis. A comprehensive study of argumentation frameworks with sets of attacking arguments. *Int. J. Approx. Reason.*, 109:55–86, 2019.

[GLP93] Giorgio Gallo, Giustino Longo, and Stefano Pallottino. Directed hypergraphs and applications. *Discret. Appl. Math.*, 42(2):177–201, 1993.

[GW13] Sarah Alice Gaggl and Stefan Woltran. The cf2 argumentation semantics revisited. *J. Log. Comput.*, 23(5):925–949, 2013.

[Jan99] Tomi Janhunen. On the intertranslatability of non-monotonic logics. *Ann. Math. Artif. Intell.*, 27(1-4):79–128, 1999.

[JV99] Hadassa Jakobovits and Dirk Vermeir. Dialectic semantics for argumentation frameworks. In Jon Bing, Andrew J. I. Jone, and Thomas F. Gordon, editors, *Proceedings of the Seventh International Conference on Artificial Intelligence and Law, ICAIL '99, Oslo, Norway, June 14-17, 1999*, pages 53–62. ACM, 1999.

[Lib14] Paolo Liberatore. Bijective faithful translations among default logics. *Journal of Logic and Computation*, 24(4):763–807, 2014.

[McC04] William McCuaig. Pólya's permanent problem. *Electron. J. Comb.*, 11(1), 2004.

[MZB07] Martin Mozina, Jure Zabkar, and Ivan Bratko. Argument based machine learning. *Artif. Intell.*, 171(10-15):922–937, 2007.

[NP06] Søren Holbech Nielsen and Simon Parsons. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In Nicolas Maudet, Simon Parsons, and Iyad Rahwan, editors, *Argumentation in Multi-Agent Systems, Third International Workshop, ArgMAS 2006, Hakodate, Japan, May 8, 2006, Revised Selected and Invited Papers*, volume 4766 of *Lecture Notes in Computer Science*, pages 54–73. Springer, 2006.

[OS11] Sebastian Ordyniak and Stefan Szeider. Augmenting tractable fragments of abstract argumentation. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 1033–1038. IJCAI/AAAI, 2011.

[Pap94] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley, Reading, MA, 1994.

[Pol17] Sylwia Polberg. *Developing the Abstract Dialectical Framework*. PhD thesis, Vienna University of Technology, Institute of Information Systems, 2017.

[Pra11] Henry Prakken. *Logical Tools for Modelling Legal Argument: A Study of Defeasible Reasoning in Law*. Springer Publishing Company, Incorporated, 1st edition, 2011.

[Sey74] Paul D. Seymour. On the two-colouring of hypergraphs. *The Quarterly Journal of Mathematics*, 25(1):303–312, 1974.

[Tur36] Alan M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2(42):230–265, 1936.

[VP00] Gerard A. W. Vreeswik and Henry Prakken. Credulous and sceptical argument games for preferred semantics. In Manuel Ojeda-Aciego, Inma P. de Guzmán, Gerhard Brewka, and Luís Moniz Pereira, editors, *Proceedings of Logics in Artificial Intelligence*, pages 239–253, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.