Die approbierte Originalversion dieser Diplom-/Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

On the Limits of Expressiveness in Abstract Argumentation Semantics:

Realizability and Signatures

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Thomas Linsbichler
Matrikelnummer 0726008

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Privatdoz. Dipl.-Ing. Dr.techn. Stefan Woltran
Mitwirkung: Dipl.-Ing. Dipl.-Ing. Dr.techn. Wolfgang Dvořák

Wien, 05.05.2013 (Unterschrift Verfasser) (Unterschrift Betreuung)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

On the Limits of Expressiveness in Abstract Argumentation Semantics:

Realizability and Signatures

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Computational Intelligence

by

Thomas Linsbichler
Registration Number 0726008

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Privatdoz. Dipl.-Ing. Dr.techn. Stefan Woltran
Assistance: Dipl.-Ing. Dipl.-Ing. Dr.techn. Wolfgang Dvořák

Vienna, 05.05.2013

____________________ (Signature of Author)

____________________ (Signature of Advisor)

Erklärung zur Verfassung der Arbeit

Thomas Linsbichler
Untere Hauptstraße 14, 3192 Hohenberg

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.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

A whole lot of people have contributed to bringing me into the situation of being about to accomplish my studies and finally writing this Master's thesis by their knowledge, friendship, music, support, motivation and wisdom. I want to thank all of them and name a few of them.

First of all I would like to thank my advisors Stefan Woltran and Wolfgang Dvořák for introducing me to the topic of abstract argumentation and for the considerable amount of time they spent on discussions and proof-reading.

I would like to thank my parents, Andrea and Ferdinand Linsbichler, for being there and making my studies possible, my girlfriend Katrin Gaupmann for her love and support, and my brother Alexander Linsbichler for proof-reading and giving me valuable advices on formal issues.

Abstract

In recent years the research field of *argumentation* has become a major topic in the study of Artificial Intelligence. In particular the formal approach of *abstract argumentation* introduced by Dung [33] has aroused much interest of research. A so-called abstract argumentation framework is a directed graph where nodes represent arguments and arrows represent conflicts between arguments, i.e. counter-arguments "attack" arguments by arrows. The question of which arguments can be accepted out of an argumentation framework is answered by *argumentation semantics*, where the outcome of applying a semantics to an argumentation framework is a set of extensions.

Surprisingly, a systematic comparison of their capability in terms of multiple extensions, and thus their expressive power in modelling multiple viewpoints with a single argumentation framework has been neglected so far. Understanding which extensions can, in principle, go together when a framework is evaluated with respect to a semantics of interest not only clarifies the "strength" of that semantics but also is a crucial issue in several applications.

The aim of the master's thesis is to study the expressiveness of the naive, stage, stable, preferred, semi-stable, and complete semantics, by characterizing their *signatures*. The signature of a semantics is defined as the set of all possible sets of extensions one can obtain from the semantics, given an arbitrary argumentation framework.

For each semantics we give necessary conditions for an extension-set to be in the signature, i.e. properties which are fulfilled by the outcomes of the semantics for each framework, as well as (corresponding) sufficient conditions for an extension-set to be in the signature, which make the extension-set *realizable* under the semantics. The thesis provides constructions of argumentation frameworks realizing extension-sets under the various semantics.

The characterizations of the signatures of the semantics give rise to a systematic comparison of their *expressiveness*. We show that, with one exception, all semantics we deal with possess different levels of expressiveness.

Finally the thesis investigates *strict realizability*, i.e. realizing an extension-set by a framework by not using additional arguments. Here we provide properties as stated above as well as impossibility results, showing that extension-sets cannot be strictly realized under certain conditions.

Kurzfassung

In den letzten Jahren hat sich das Forschungsfeld der Formalen *Argumentation* als eine Hauptströmung im Bereich der Künstlichen Intelligenz etabliert. Allen voran löste Dungs Ansatz der *Abstract Argumentation* [33] großes Forschungsinteresse aus. Ein sogenanntes Abstract Argumentation Framework (AF) ist ein gerichteter Graph, dessen Knoten Argumente und dessen gerichtete Kanten Konflikte zwischen diesen Argumenten repräsentieren. Wird ein Argument als Gegenargument zu einem anderen Argument angesehen, so wird dies durch eine gerichtete Kante vom ersten zum zweiten Argument dargestellt. Die Frage, welche Argumente eines AFs gemeinsam akzeptiert werden können, wird durch *Semantiken* beantwortet, wobei das Ergebnis der Anwendung einer Semantik auf ein AF als eine Menge von Extensionen bezeichnet wird.

Überraschenderweise wurde bisher ein systematischer Vergleich der Ausdruckskraft verschiedener Semantiken, und folglich deren Fähigkeit, verschiedene Standpunkte eines einzelnen AFs darzustellen, vernachlässigt. Das Wissen über die Tatsache, welche Extensionen gemeinsam das Ergebnis der Anwendung einer Semantik auf ein AF darstellen können, gibt nicht nur Aufschlüsse über die *Ausdrucksstärke* einer Semantik. Vielmehr ist es auch von großem Vorteil für eine Fülle von Anwendungen.

Das Ziel dieser Diplomarbeit ist die Untersuchung der Ausdrucksstärke der naive, stage, stable, preferred, semi-stable und complete Semantik. Dies wird durch die Charakterisierung der *Signaturen* dieser Semantiken bewerkstelligt. Die Signatur einer Semantik ist durch die Menge aller möglicher Mengen an Extensionen, die durch die Anwendung der Semantik auf ein beliebiges AF erlangt werden können, definiert.

Die Arbeit definiert für jede Semantik notwendige Bedingungen für eine Menge an Extensionen, um Teil der Signatur zu sein, also Eigenschaften, die von jedem Ergebnis der Semantik erfüllt sind, sowie (entsprechende) hinreichende Bedingungen einer Menge an Extensionen, um Teil der Signatur zu sein, also Eigenschaften, welche die *Realisierbarkeit* der Menge an Extensionen durch die Semantik bezeugen. Weiters beinhaltet die Diplomarbeit Konstruktionen von AFs, welche die gegebene Menge an Extensionen durch die jeweilige Semantik realisieren.

Die Charakterisierung der Signaturen der Semantiken ermöglicht einen systematischen Vergleich derer Ausdruckskraft. Die Ergebnisse dieser Arbeit zeigen, dass, mit einer einzigen Ausnahme, alle behandelten Semantiken verschiedene Grade an Ausdrucksstärke aufweisen.

Schließlich untersucht die Arbeit *Strikte Realisierbarkeit*, d.h. Realisierbarkeit von Extensionsmengen durch AFs, welche keine zusätzlichen Argumente verwenden. Dazu enthält die Arbeit ähnliche Bedingungen wie oben bereits beschrieben, sowie Resultate, welche die Unmöglichkeit der strikten Realisierbarkeit bestimmter Extensionsmengen bezeugen.

Contents

CHAPTER 1

Introduction

1.1 Argumentation and Artificial Intelligence

The theory of argumentation comes within the oldest disciplines of philosophy by dating back to ancient Greek philosophers and rhetoricians, such as Aristotle. It is concerned with identifying the requirements that make an argument correct. As an interdisciplinary research area, argumentation gives answers to "how assertions are proposed, discussed and resolved in the context of issues upon which several diverging opinions may be held" [12]. It studies mechanisms to distinguish "legitimate" arguments having reasonable support from "flawed" arguments. Generally speaking, it addresses how conclusions can be made by logical reasoning.

This gives rise to the use of formal logic and mathematical proof which already suggests the connection of argumentation theory and the theory of computation. Actually, argumentation has become a major topic in Artificial Intelligence (AI). A very popular and heavily studied formalism is abstract argumentation as introduced by Dung in [33]. Argumentation has a big variety of applications such as legal reasoning [53], dialogue games [47], decision support systems [1] or multi-agent systems [58]. Also some notable systems incorporating argumentation theory such as the Parmenides system [2, 27], the argumentation toolbox of the IMPACT project[1] and the ASPIC+ framework [51, 54] have emerged.

Being a vital research topic, argumentation can be found as a keyword at all major AI conferences. The biennial conference "Computational Models of Argument (COMMA)"[2] is solely dedicated to argumentation, just as several workshops and the journal "Argument and Computation"[3]. The text books [55] and [14] give an overview of the studies on argumentation theory and [12] summarizes recent work on argumentation in AI.

[1] http://www.policy-impact.eu/
[2] http://www.comma-conf.org/
[3] http://www.tandfonline.com/toc/tarc20/current

The Argumentation Process

Put simply, argumentation consists of extracting arguments out of some knowledge base or dialogue, identifying conflicts between those arguments and finally finding a set of acceptable arguments which is reasonably consistent based on the identified conflicts. This can be placed in the context of a single agent arguing about decisions based on his knowledge or in the field of multi-agent domains where the problem of diverging opinions has to be resolved. A famous example of the latter case is the Liar Paradox:

Three persons A, B and C make the following statements:

- A: “I am a liar”.
- B: “Person A is a liar”.
- C: “Person B is a liar”.

The question that emerges is, which of the persons says the truth.

Having three single statements as knowledge base it is easy to see that the statements coincide with the arguments. Let us denote the statements of persons A, B and C as arguments a, b and c. Also the conflicts are immediate. While argument a stands in conflict with itself, argument b defeats a and c defeats b. Reasoning about the validity of the given statements, i.e. finding the set of acceptable arguments, is a non-trivial task. One could argue that accepting argument b is reasonable as well as accepting argument c, since both do not stand in conflict for themselves. On the other hand one could deny the acceptance of b as it is defeated by c without having a counter-argument. We will come back to this issue in Section 2.2.

The following, more extensive description of the argumentation process is inspired by [23] and taken out of [36]:

1. Start with or build the knowledge base
2. Extract arguments out of the knowledge base
3. Identify conflicts between arguments
4. Abstract from the internal structure of the arguments
5. Resolve conflicts between arguments and select acceptable subsets of arguments
6. Draw conclusions

For none of the steps there is a uniform way to accomplish the task. The knowledge base can consist of a set of logical formulas or just a collection of informal opinions. When extracting arguments out of the knowledge base (Step 2) there can be dispute on how to choose the granularity of arguments. Also for the identification of conflicts (Step 3), numerous different approaches can be applied. Recent work on the instantiation of argumentation systems tackles these issues [23, 54]. However, the main contributions of this thesis are in the field of Dung’s abstract argumentation frameworks [33], which form an abstraction obtained from Step 4 and are a playground for various reasonable ways of detecting acceptable sets of arguments (Step 5); a process, which is described by argumentation semantics. The outcome of argumentation semantics is then used to draw conclusion (Step 6) by considering the actual contents of the accepted arguments.

Abstract Argumentation

Abstract argumentation frameworks (AFs for short), as introduced in Dung's seminal paper [33], conceal the concrete contents of arguments and only consider the relation between them. In fact, AFs are just directed graphs, where nodes represent the arguments and edges represent conflicts between the arguments. The popularity of these frameworks is based on their ability to preserve simplicity and powerfulness at the same time. Argumentation semantics describe formal methods to identify sets of acceptable arguments out of a given AF, where one such set of acceptable arguments is called an extension. The last 15 years have seen an enormous effort to design, compare, and implement different semantics for Dung's abstract argumentation frameworks. While several semantics are already defined in [33], many others have been proposed by various authors in the following years [4, 8, 21, 34, 60]. The interested reader finds an overview in [3].

The fact that all steps in the argumentation process are, in general, intractable gives rise to complexity analysis. For argumentation semantics this has been well studied. For an overview on complexity results on argumentation semantics we refer to [35, 36].

Several implementations have been proposed in order to compute the outcomes of argumentation semantics. These implementations can be grouped into two categories, the ones following a reduction approach and the ones following a direct approach. The idea of the reduction approach is to exploit existing efficient software, originally developed for other purposes. Herein come, upon others, the SAT-based approach [13], the QBF-based approach [41], and the ASP-based approach [40, 59]. The labelling approach [32, 50] and dynamic programming approaches [37] come within the big variety of direct approaches, where algorithms are developed from scratch.

1.2 Limits of Expressiveness in Abstract Argumentation Semantics

As already mentioned, argumentation semantics are a topic of heavy research. Surprisingly, a systematic comparison of their capability in terms of multiple extensions, and thus their power in modelling multiple viewpoints with a single AF has been neglected so far. Understanding which extensions can, in principle, go together when a framework is evaluated with respect to a semantics of interest not only clarifies the "strength" of that semantics but also is a crucial issue in several applications.

In this work, we close this gap by studying the *signatures* of several important semantics[4] namely naive, preferred, complete, semi-stable, stage, and stable semantics [24, 33, 60]. By the

[4] We will also study the signatures of conflict-free and admissible sets, which are no semantics in the narrow sense.

term signature for a semantics σ we understand the set

$$\Sigma_\sigma = \{\sigma(F) \mid F \text{ is an AF}\},$$

where $\sigma(F)$ is the set of extensions we get by applying the semantics σ to the AF F. The signature of a semantics is the collection of extension-sets which can be obtained by the semantics. Knowing whether a set of extensions $\mathcal{S}$ is contained in Σ_σ provides information on whether one can find an AF where the application of the semantics σ has $\mathcal{S}$ as its result. We will provide constructions for AFs where this is the case.

Characterizing the signatures and by that finding simple criteria to decide whether a set of extensions $\mathcal{S}$ is contained in Σ_σ for different semantics σ is essential in many aspects. In what follows, we highlight a few of them.

Expressiveness of Semantics Baroni et. al. have done an extensive comparison of semantics by means of different properties (see [5–7]). So far these properties mostly focused on the aspects of a single extension $S \in \mathcal{S}$ rather than on a set $\mathcal{S}$ thereof. An obvious exception is incomparability[5] (the sets in $\mathcal{S}$ are not proper subsets of each other). However, as we will see, all of the standard semantics put additional (yet different) requirements on $\mathcal{S}$ in order to be contained in the signature. The signatures also tell us about how much disagreement in the shape of [16] can be obtained from certain semantics.

By characterizing the signatures, we will get a picture of the expressiveness of various semantics, i.e. the levels of "disagreement" semantics can express. The comparison of signatures of semantics also gives information whether two semantics can express the same extension-sets or one semantics is more expressive then the other. Such a comparison has been done by recent studies of intertranslatability issues [38, 39], where signatures of semantics are put in relation to each other[6]. More precisely, if there is a translation such that extensions in terms of semantics σ of the transformed AF coincide with the extensions in terms of a different semantics θ of the original AF, then θ is at least as expressive as σ, that is $\Sigma_\sigma \subseteq \Sigma_\theta$ in our terms. These results, however, do not tell us anything about the actual contents of Σ_σ and Σ_θ. The results in this thesis will give a full characterization of the contents of Σ_σ where σ stands for the naive, preferred, complete, semi-stable, stage, and stable semantics.

Similar work has been done in other nonmonotonic formalisms. Studying the expressiveness of KR formalisms is crucial in case not all possible sets of intended models have a syntactical counterpart. In terms of classical logic, it is obvious that, for any given set I of interpretations, there exists a propositional formula having I as its models. When it comes to fragments of logic, the situation changes. We refer to [62] for a more recent study of this topic, also in terms of how formulas can be constructed from a given set I of interpretations. As we will point out in the next paragraph, knowing the expressiveness of the language under consideration is important in the area of belief change. In this area recent work has focussed on revision in fragments of logic and it is unsurprising that restricting expressiveness has to be taken into consideration [30, 31]. Finally, characterizations of possible outcomes have also been studied in the area of logic programming under the answer-set semantics, see e.g. [42].

[5] Incomparability is called *I-maximality* by Baroni et. al.

[6] The term signature is not used in these papers yet.

Model-based Revision The results of this thesis are also important for constructing AFs. Indeed, knowing whether a set $\mathcal{S}$ is contained in Σ_σ is a necessary condition which should be checked before actually looking for an AF F which *realizes* $\mathcal{S}$ under σ, meaning that applying σ to F yields $\mathcal{S}$, i.e. $\sigma(F) = \mathcal{S}$. This is of high importance when dynamic aspects of argumentation are considered, as, for instance, done in [15, 44]. As an example, suppose a framework F possesses a set $\mathcal{S}$ as its σ-extensions and one asks for an adaptation of the framework F such that its σ-extensions are given by $\mathcal{S} \cup \{E\}$, i.e. one extension shall be added. Before starting to think on how the adapted framework could look like it is obviously crucial to know whether an appropriate framework exists at all, i.e. whether $(S \cup \{E\}) \in \Sigma_\sigma$. As an example consider the preferred semantics, which will be formally introduced in Section 2.2. For the moment, the thing that is important to know is that extensions of preferred semantics are always incomparable and admissible. There exists some framework F which has the set $\mathcal{S} = \{\{a,b\},\{a,c\}\}$ as its preferred extensions. As preferred extensions are incomparable, obviously an adaption of the framework, such that the preferred extensions of the new framework are $\mathcal{S}' = \{\{a\},\{a,b\},\{a,c\}\}$, is not possible. However, we will see later that there is also no adapted framework F' having $\mathcal{S}'' = \{\{a,b\},\{a,c\},\{b,c\}\}$ as its preferred extensions, although the elements of $\mathcal{S}''$ are incomparable. We will see that there are other cases where model-based belief revision is not possible due to other necessary properties of extension-sets.

Instantiation-based Argumentation We also stress the connections between the central topic of this paper and work on instantiation-based argumentation [23]. In such studies the concept of *rationality postulates* plays an important role as does the underlying principle of evaluating argumentation semantics in generic terms. In other words, attention is focussed on properties in common (for example conflict-freeness, reinstatement) rather than on aspects that highlight distinctions (e.g. in stable extensions the potential for the set of extensions to be empty). In this thesis our concern is, also, with characteristics shared by classes of extension-based semantics with respect to the question of realizability. Our results will show that the rationality postulates cannot be met by the naive semantics under any instantiation.

Pruning of Search-space There are numerous implementations for computing the extensions of argumentation semantics [13, 32, 37, 40, 41, 50, 59]. In general, each of these implementations builds up some search-space, from which it detects the desired extensions in an algorithmic way. The most naive implementation would, given an AF F with the sets of arguments A, iterate through all extension-candidates $S \subseteq A$ and test whether S is a valid extension for a given semantics σ. The search-space of this procedure is then the set of all extension-candidates, i.e. $\{S \mid S \subseteq A\}$. Being aware of certain properties all extension-sets of a given semantics fulfill, gives the possibility of pruning the search-space. We show this on another example with preferred extensions, keeping in mind that the preferred extensions of any AF are incomparable.

Consider an AF F having the set A as arguments and a procedure P, which determines all preferred extensions of F. Now assume P has detected $\{a,b,c\} \subseteq A$ as a preferred extension. By incomparability of preferred extension the search-space of P can be pruned in a way that all A^- and A^+ with $A^- \subset \{a,b,c\} \subset A^+$ are removed from the set of extension-candidates. With the knowledge of other necessary properties of extension-sets under certain semantics, more involved pruning of search-space is possible.

1.3 Related Work

Many references to related work have already been given in Section 1.2. Here, we highlight the research closest to the results of this thesis.

Comparison of semantics A systematic comparison of argumentation semantics can be found in [5–7]. The authors introduce properties such as *I-maximality*, *(strong) admissibility*, *(weak) reinstatement*, *cf-reinstatement* and *directionality* and show whether the common semantics fulfill these properties or not. However, except I-maximality, these properties refer to single extensions instead of multiple extensions. Their work contributes to the direction of research, which aims at developing the "right" semantics. The outcome of this direction of research has been called a "plethora of argumentation semantics" by G. Simari[7].

Intertranslatability The work which is closest to our investigations are recent studies on intertranslatability issues [38,39,57]. The authors describe translations between the most common argumentation semantics, where, given an AF F, a translation from semantics σ to semantics θ builds another AF F' such that $\sigma(F)$ and $\theta(F')$ stand in a certain relation to each other. If $\sigma(F) = \theta(F')$ holds for every given F, the authors talk about exact translations. Such an exact translation from σ to θ puts the signatures Σ_σ and Σ_θ into relation in a way that θ is at least as expressive as σ. While [39] focusses on computationally efficient translations and gives results which show the impossibility of such efficient translations between certain semantics, [38] tries to give a full picture of intertranslatability by not restricting the translations.

Enforcement Another work which is related to the topic of realizability and signatures is recent research on enforcement in abstract argumentation [9,10]. The authors give both possibility and impossibility results related to the problem of enforcing a desired set of arguments. This comes, to some extent, close to investigating sufficient conditions in order to provide realizability of extension-sets.

1.4 Main Contributions

For the results in this thesis, the semantics of our interest are the naive, preferred, complete, stable, semi-stable and stage semantics as well as conflict-free and admissible sets. The ultimate goal is to characterize the signatures for each of these semantics, i.e. which sets of extensions have an AF as their syntactical counterpart under a certain semantics. More specifically, the main contributions of this thesis are:

[7]During the presentation of [48] at COMMA 2006.

- We categorize extension-sets by defining certain properties and show for each semantics of our interest, which properties are fulfilled by each set of extensions obtained by applying the semantics to any AF.
- We introduce and study realizability: the issue of whether it is possible to find, given a set of extensions $\mathcal{S}$ and a semantics σ, an AF having exactly $\mathcal{S}$ as its σ-extensions.
- For each of the semantics we describe how to build an AF upon a realizable set of extensions. We define constructions of canonical frameworks (for each semantics σ), which realize given extension-sets under σ. In order to formalize these constructions we make use of propositional logic to describe certain dependencies between arguments.
- We also touch upon optimization issues and strengthen the concept of realizability in such a way that we want to find an AF F which is solely built from arguments occurring in $\mathcal{S}$ and delivers $\sigma(F) = \mathcal{S}$ (hence, no additional arguments to express $\mathcal{S}$ are required). We show that for naive semantics each $\mathcal{S} \in \Sigma_{naive}$ can be strictly realized, while this is not the case for the other semantics.
- We identify characteristics of signatures in the main extension-based approaches. Verifying whether a set of extensions is in the signature tells us about whether this set of extensions is realizable. Further we relate the signatures of the different semantics to each other, and therefore obtain theorems concerning the expressiveness of semantics.

Methodology Being located in the field of theoretical computer science the main methodological approach of the thesis is to use formal methods to prove or disprove claims. Methods of discrete mathematics give a formal basis as well as set-theory and propositional logic. Further fundamental theorems on semantics as given in [33] and intertranslatability results [38, 39] are used to achieve results on realizability and signatures.

1.5 Organization of the Thesis

The remainder of the thesis will be organized as follows:

- In Chapter 2 the main concepts of abstract argumentation, argumentation semantics and propositional logic will be introduced.
- Chapter 3 introduces, after giving some basic definitions, several properties of extension-sets. Then for every semantics of our interest certain properties are proven to be fulfilled by every extension-set obtained from the semantics.
- In Chapter 4 we will examine for each of the semantics, which properties an extension-set has to fulfill in order to guarantee that one can find an AF realizing the extension-set under the given semantics. Moreover, we will provide formal descriptions of how to construct such AF for each of the semantics. Finally we will introduce the concept of strict realizability and show, that with the exception of conflict-free sets and naive semantics, general and strict realizability do not coincide in general.

- Chapter 5 assembles the results of Chapters 3 and 4 and gives the formal characterizations of the signatures. It further relates the signatures to each other and by that gives a picture of different levels of expressiveness of the semantics. Finally we show the connection of the results of this thesis to recent work on intertranslatability [38, 39].
- In Chapter 6 we will conclude by summarizing our results, pointing out the implications resulting from them and giving an outlook to possible future research directions.

CHAPTER 2

Preliminaries

This chapter aims at introducing the main concepts of abstract argumentation frameworks, argumentation semantics, and propositional logic. This introduction makes no claim to be exhaustive, but covers all concepts needed in order to propose and prove the main results of this thesis. Abstract argumentation frameworks were first introduced in Dung's seminal paper [33], proper overviews can be found in [12] and [55]. Argumentation semantics have been introduced by several authors over the years [4, 8, 21, 33, 60], an overview is given in [3]. For an exhaustive work on propositional logic we refer to [43].

2.1 Abstract Argumentation Frameworks

Throughout this thesis we use $\mathcal{A}$ as a countably infinite set of arguments. We may just call $\mathcal{A}$ a set of arguments, implicitly meaning a countably infinite one.

Definition 2.1. An *argumentation framework* (AF for short) is a pair $F = (A, R)$ where $A \subseteq \mathcal{A}$ is a finite set of arguments and $R \subseteq A \times A$ is the attack relation. Given an AF $F = (A, R)$ we use A_F to denote the set of arguments A and R_F to denote the set of attacks R. The collection of all AFs (over $\mathcal{A}$) is given as $AF_{\mathcal{A}}$.

Note that basically, the set of arguments of an AF does not have to be finite. However, for practical reasons, we restrict ourselves to finite AFs. We will give some notes on infinite AFs at the end of Section 2.2.

Definition 2.2. Given an AF $F = (A, R)$ we say a attacks b, or b is defeated by a if $(a, b) \in R$. We write $a \mapsto_R b$ or, if no ambiguities arise, just $a \mapsto b$ for $(a, b) \in R$.

Definition 2.3. Given an AF $F = (A, R)$, sets of arguments $S, T \subseteq A$ and an argument $a \in A$, we may use the following shortcuts:

- $S \mapsto_R a$ if $\exists s \in S : \; s \mapsto_R a$.

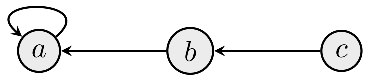

Figure 2.1: AF F_l of the liar paradox

- $a \mapsto_R S$ if $\exists s \in S:\ a \mapsto_R s$.
- $S \mapsto_R T$ if $\exists s \in S\ \exists t \in T:\ s \mapsto_R t$.

Again, if no ambiguities arise, we may drop subscript R in $\mapsto_R$.

In Chapter 1 we discussed the necessity of studying the theory of argumentation on the liar paradox. The following example shows the liar paradox as an AF.

Example 2.1. Consider the liar paradox introduced in Chapter 1. The corresponding AF is given by $F_l = (A, R)$ where

$$A = \{a, b, c\} \text{ and}$$

$$R = \{(a, a), (b, a), (c, b)\}.$$

The graphical representation of this framework is given in Figure 2.1.

As one can see, the graphical version of an AF is a directed graph, where the nodes are given by the arguments and the arrows are given by the attacks. An arrow from a to b in the graph of some AF F corresponds to $(a, b) \in R_F$.

In the remainder of this thesis we may, especially for AFs of large size, omit the conventional notation and only give the graphical representation.

Example 2.2. In the remainder of this chapter we will make heavy use of the AF $F_s = (A, R)$ where

$$A = \{a, b, c, d, e\} \text{ and}$$

$$R = \{(a, b), (c, b), (c, d), (d, c), (d, e), (e, e)\}.$$

To show the behaviour of the different semantics we will use this framework as our first reference. Figure 2.2 shows its graphical representation.

One central concept when it comes to evaluating the justification of an argument is the concept of *defense*.

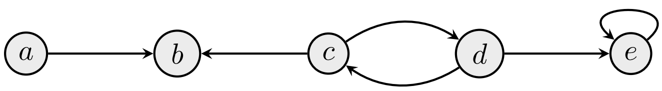

Figure 2.2: Graphical representation of the AF $F_s = (\{a, b, c, d, e\},\ \{(a, b),\ (c, b), (c, d), (d, c), (d, e), (e, e)\})$

Definition 2.4. Given an AF $F = (A, R)$, sets of arguments $S, T \subseteq A$ and an argument $a \in A$, we say

- a is *defended* by S (in F) if for each $b \in A$ it holds: if $b \mapsto_R a$ then $S \mapsto_R b$,
- T is defended by S (in F) if each $t \in T$ is defended by S (in F), and
- S defends a from T (in F) if for all $b \in T$ it holds: if $b \mapsto_R a$ then $S \mapsto_R b$.

The function $\Phi_F : 2^A \to 2^A$ such that $\Phi_F(S) = \{a \mid a \text{ is defended by } S \text{ in } F\}$ is called the *characteristic function* of F.

In Dung's seminal paper [33] the author uses the term *acceptability* and so "a is acceptable with respect to S" is used instead of "a is defended by S". In this thesis we will stick to the term "defended by" since this seems to correspond to the intuition of having all attackers attacked.

The following technical lemma will be useful for the proofs of several propositions throughout this thesis.

Lemma 2.1. *Given an* AF $F = (A, R)$ *and two sets of arguments* $S, T \subseteq A$*, the following holds: If* S *is defended by itself in* F *and* T *is defended by itself in* F*, then* $S \cup T$ *is defended by itself in* F*.*

Proof. To the contrary assume that $S \cup T$ is not defended by itself in F. Then there exists a $b \in A$ with $b \mapsto (S \cup T)$ such that $(S \cup T) \mapsto b$ does not hold. We can distinguish between two cases, either $b \mapsto S$ or $b \mapsto T$. Consider $b \mapsto S$. Since $(S \cup T) \mapsto b$ does not hold, $S \mapsto b$ does not hold either. Therefore S is not defended by itself in F which is a contradiction to the assumption. The other case behaves symmetrically. □

The fact that the union of two sets of arguments, where each of them defends itself, defends itself too will be useful later on in this thesis. We will come back to that in Section 2.2.

We go on with some more definitions.

Definition 2.5. Given an AF $F = (A, R)$ and a set of arguments $S \subseteq A$, the union of S with the set of attacked arguments, or for short the *range* of S with respect to R, denoted by S_R^+, is defined as $S \cup \{t \mid S \mapsto_R t\}$.

Definition 2.6. Given two AFs $F_1 = (A_1, R_1)$ and $F_2 = (A_2, R_2)$ the *union* $F_1 \cup F_2$ is given by

$$F_1 \cup F_2 = (A_1 \cup A_2, R_1 \cup R_2).$$

We examine the definitions given so far on the AFs given in Example 2.1 (see Figure 2.1) and Example 2.2 (see Figure 2.2).

Example 2.3. Consider the AF $F_l = (A, R)$ given by Figure 2.1. We observe that

- c is defended by $\{c\}$ in F_l,
- a is defended by $\{a, c\}$ in F_l,

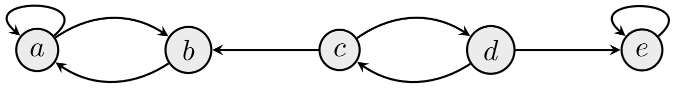

Figure 2.3: Union of the AFs F_l from Figure 2.1 and F_s from Figure 2.2

- $\Phi_{F_l}(\{c\}) = \{c\}$, $\Phi_{F_l}(\{a\}) = \Phi_{F_l}(\{b\}) = \emptyset$, and
- $\{b\}^+_R = \{a, b\}$.

Example 2.4. Consider the AF $F_s = (A, R)$ given by Figure 2.2. We observe that

- $\{d\}$ is defended by itself in F_s,
- $\{c, e\}$ is defended by itself in F_s,
- $\{d\}$ defends b from $\{c\}$ in F_s (but not from a),
- $\Phi_{F_s}(\{d\}) = \{a, d\}$, and
- $\{c\}^+_R = \{b, c, d\}$.

Example 2.5. Let F_l be the AF given by Figure 2.1 and F_s the AF given by Figure 2.2. The union $F_1 \cup F_2$ of these two frameworks is the AF in Figure 2.3.

2.2 Argumentation Semantics

Given an argumentation framework there is no, and cannot be, consensus about which sets of arguments collectively form a reasonable position. Different applications and fields of application may require different rules how to identify such sets. An *abstract argumentation semantics* is a formal method to identify conflict outcomes for any argumentation framework. It defines, which sets of arguments can survive together the conflict given by an AF. In this thesis we deal with the *extension*-based approach, where a set of arguments accepted by a semantics is called an extension. For an excellent recent overview we refer to [3].

Definition 2.7. An (extension-based) semantics for abstract argumentation frameworks is a function $\sigma : AF_{\mathcal{A}} \to 2^{2^{\mathcal{A}}}$ mapping each AF F to a set of extensions $\sigma(F) \subseteq 2^{A_F}$. A semantics σ is called a *unique status semantics* if for each F, $\|\sigma(F)\| = 1$, otherwise it is called a *multiple status semantics*.

Dung's seminal paper [33] gives, besides the formal introduction of AFs, definitions of the *stable*, *preferred*, *complete* and *grounded* semantics. The *stage* semantics were introduced in [60], where also the *semi-stable* semantics were implicitly mentioned. The actual term "semi-stable" was coined later in [21].

The most basic concept shared by all argumentation semantics in the literature is the concept of *conflict-freeness*. Arguments going together into an extension should not be in conflict with respect to the attack relation.

Definition 2.8. Given an AF $F = (A, R)$, a set $S \subseteq A$ is called *conflict-free* in F if there are no $a, b \in S$ such that $a \mapsto_R b$. The collection of all conflict-free sets is denoted as $cf(F)$.

Note that by this definition self-attacking arguments can never be in any conflict-free set. As all semantics (presented in this thesis) share the concept of conflict-freeness, self-attacking arguments cannot be part of any extension of any semantics. Since $\emptyset$ is conflict-free by definition, there is always at least one conflict-free set, given an arbitrary AF.

Example 2.6. Take into account the AF F_s from Figure 2.2. The conflict free sets in F_s are $cf(F_s) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{a,c\}, \{a,d\}, \{b,d\}\}$. Note that as e is self-attacking, i.e. $(e,e) \in R_{F_s}$, there is no $E \in cf(F_s)$ with $e \in E$. Further consider the arguments c and d. As $(c,d) \in R_{F_s}$ (and $(d,c) \in R_{F_s}$) there is no $E \in cf(F_s)$ with $\{c,d\} \subseteq E$.

Having introduced conflict-freeness, we turn to another basic concept of argumentation semantics, *admissibility*. The idea is, that each element of an admissible set has a kind of justification in form of being defended by the set.

Definition 2.9. Given an AF $F = (A, R)$, a set $S \subseteq A$ is an *admissible* set in F if S is conflict-free in F and S is defended by itself in F, i.e. for all $b \in A$ where $b \mapsto_R a$ there is some $s \in S$ such that $s \mapsto_R b$. The collection of all admissible sets is denoted as $adm(F)$.

Note that again, there is always at least one admissible set, as $\emptyset$ is not only conflict-free, but also defended by itself, and therefore $\emptyset \in adm(F)$ for any AF F.

An alternative definition can be given via the characteristic function.

Corollary 2.2. *Given an* AF $F = (A, R)$*, a set* $S \subseteq A$ *is an admissible set in* F *if* S *is conflict-free in* F *and* $S \subseteq \Phi_F(S)$.

Example 2.7. Again consider the AF F_s from Figure 2.2. The admissible sets are $adm(F_s) = \{\emptyset, \{a\}, \{c\}, \{d\}, \{a,c\}, \{a,d\}\}$. Take into account the conflict-free set $\{b,d\}$. Argument d defends itself and b from attacker c, but b is not defended from attacker a. Therefore $\{b,d\}$ is not admissible. On the other hand consider the set $\{c,e\}$, which is defended by itself (see Example 2.4). As e attacks itself, $\{c,e\}$ is not conflict-free and therefore not admissible either.

It is easy to see by definition, that each admissible set is a conflict-free set. Another connection between admissible and conflict-free sets can be seen in Dung's fundamental lemma [33]:

Lemma 2.3. *Let* S *be an admissible set of arguments in an* AF F*, and* a *and* a' *be arguments which are defended by* S *in* F*. Then*

1. $S' = S \cup \{a\}$ *is admissible in* F*, and*
2. a' *is defended by* S' *in* F.

By Lemma 2.1 we know that the union of two sets of arguments, where each of them defends itself, defends itself too. Together with Lemma 2.3 it follows that also the union of two admissible sets is admissible itself.

Although not treated as semantics in the narrow sense in the literature, we will use the terms "conflict-free semantics" and "admissible semantics" at some points throughout this thesis, since when dealing with realizability and signatures we do not make a difference between conflict-free and admissible sets and the other, traditional, semantics.

We will come back to admissibility but first put it aside and introduce another semantics.

Definition 2.10. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *naive* extension in F if S is conflict-free in F and there is no $T \in \mathit{cf}(F)$ with $T \supset S$. The collection of all naive extensions is denoted as $\mathit{naive}(F)$.

So naive extensions are the conflict-free sets, which are maximal with respect to set-inclusion. Therefore they may be referred to as maximal conflict-free sets. It is easy to see that as for each AF F there is always at least one conflict-free set, also the set of naive extensions is non-empty.

Further note that as the definition of naive semantics does not make use of the concept of defense and only forces the arguments of an extension to be conflict-free (and $\subseteq$-maximal of course), the direction of attacks in an AF does not matter. It only matters if there is an attack between two arguments, but not which direction this attack has or if it is a symmetric attack. Of course this also holds for conflict-free sets.

We look at another example.

Example 2.8. Consider the AF F_s from Figure 2.2. The naive extensions are $\mathit{naive}(F_s) = \{\{a, c\}, \{a, d\}, \{b, d\}\}$. As one can easily see these are just the $\subseteq$-maximal sets of $\mathit{cf}(F_s)$.

By Definition 2.10 it is clear that each naive extension is a conflict-free set.

We now turn to the preferred semantics. Like the naive semantics, the preferred semantics follows the idea of maximizing the accepted arguments. The difference is that the preferred semantics takes into account maximal admissible sets, not maximal conflict-free sets.

Definition 2.11. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *preferred* extension in F if S is admissible in F and there is no $T \in \mathit{adm}(F)$ with $T \supset S$. The collection of all preferred extensions in F is denoted as $\mathit{pref}(F)$.

We have seen that, given an arbitrary AF, there is always at least one admissible set. Therefore also a preferred extension exists.

Example 2.9. Take into account the AF F_s from Figure 2.2. The preferred extensions are $\mathit{pref}(F_s) = \{\{a, c\}, \{a, d\}\}$. For each of the admissible sets $\{a\}, \{c\}, \{d\}$ there is some $E \in \mathit{adm}(F_s)$ such that E is a superset. Therefore these sets are no preferred extensions.

We now turn to complete semantics. While admissibility requires some justification for each argument, complete semantics make one further step by permitting to abstain from arguments only if there are no good reasons to do otherwise. More concrete, it enforces each argument which is defended by some potential extension to be in the extension too. So the complete semantics can be seen as "a strengthening of the basic requirements enforced by the idea of admissibility" [3].

Definition 2.12. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *complete* extension in F if S is admissible in F and for each $a \in A$ that is defended by S in F it holds that $a \in S$. The collection of all complete extensions in F is denoted as $\mathit{comp}(F)$.

Again an alternative definition can be given by the characteristic function.

Corollary 2.4. *Given an* AF $F = (A, R)$*, a set* $S \subseteq A$ *is a complete extension if* S *is conflict-free and* $S = \Phi_F(S)$*.*

As one can see from this alternative definition, each complete extension is a fixed point of the characteristic function Φ.

We look at an example:

Example 2.10. Again consider the AF F_s from Figure 2.2. The set of complete extensions is $\mathit{comp}(F_s) = \{\{a\}, \{a, c\}, \{a, d\}\}$. For all $E \in \mathit{comp}(F_s)$ it holds that $E = \Phi_{F_s}(E)$. It does not hold for $\{c\}$ and $\{d\}$ since a, having no attacker, is always defended and therefore $a \in \Phi_{F_s}(\{c\})$ and $a \in \Phi_{F_s}(\{d\})$.

It is easy to see by Definition 2.12 that each complete extension it an admissible set. Another relation is given in [33]:

Proposition 2.5. *Given an arbitrary* AF $F = (A, R)$*, each preferred extension is a complete extension.*

In can be seen in Example 2.10 that the reverse does not hold, as $\{a\}$ is a complete extension but not a preferred extension (see Example 2.9).

By Proposition 2.5 it follows immediately that the set of complete extensions is non-empty.

The following lemma will be useful later in this thesis.

Lemma 2.6. *Given an* AF $F = (A, R)$ *and some admissible set* $E \in \mathit{adm}(F)$*, there exists a unique complete extension* $E' \in \mathit{comp}(F)$ *with* $E' \supseteq E$ *such that for all* E'' *with* $E' \supset E'' \supset E$ *it holds that* $E'' \notin \mathit{comp}(F)$*.*

Proof. Consider some $E \in \mathit{adm}(F)$. If $E \in \mathit{comp}(F)$ then $E' = E$. If $E \notin \mathit{comp}(F)$ then there exists a set of arguments $D \subseteq (A \setminus E)$ which is defended by E. By Lemma 2.3, $E' = E \cup D$ is admissible, i.e. $E' \in \mathit{adm}(F)$. If $E' \in \mathit{comp}(F)$ we are done. If $E' \notin \mathit{comp}(F)$ then again there exists a set of arguments $D' \subseteq (A \setminus E')$ which is defended by E'. By Lemma 2.3, $E'' = E' \cup D'$ is admissible, i.e. $E'' \in \mathit{adm}(F)$. As A is finite, we can show by induction that at some point we get some unique $\bar{E} \supset E$ which is admissible and it holds that $\bar{E} \in \mathit{comp}(F)$ and for all $\widetilde{E}$ with $\bar{E} \supset \widetilde{E} \supset E$, $\widetilde{E} \notin \mathit{comp}(F)$. □

So far we have presented solely multiple status semantics. The next, namely the grounded semantics, is a unique status semantics, i.e. for each AF there is exactly one grounded extension (cf. Definition 2.7). Considering each complete extension as a reasonable position, the grounded extension is the set of arguments shared by all these reasonable positions.

Definition 2.13. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *grounded* extension if $S \in \mathit{comp}(F)$ and there is no $T \in \mathit{comp}(F)$ with $T \subset S$. The collection of all grounded extensions in F is denoted as $\mathit{grd}(F)$[1].

As the set of complete extensions is non-empty and the grounded extension is just the $\subseteq$-minimal complete extension, the grounded extension always exists. Also each grounded extension is a complete extension.

An alternative definition of the grounded extension is by the least fixed point of the characteristic function Φ.

Corollary 2.7. *Given an* AF $F = (A, R)$*, the grounded extension is given by* $\bigcup_{i=1,\ldots,\infty} \Phi^i_F(\emptyset)$*, where* $\Phi^1_F(\emptyset) = \Phi_F(\emptyset)$ *and* $\Phi^i_F(\emptyset) = \Phi_F(\Phi^{i-1}_F(\emptyset))$ *for* $i > 1$*.*

In other words the grounded extension is obtained by the following algorithm, given some AF $F = (A, R)$ and starting with $S = \emptyset$:

1. Put each argument $a \in A$ not attacked in F into S. If no such a exists, return S.
2. Remove from F all arguments in S^+_F together with all adjacent attacks and continue with step 1.

Example 2.11. First consider AF F_s from Figure 2.2. As a is not attacked in F_s, $\Phi_{F_s}(\emptyset) = \{a\}$. Since $\{a\}$ does not defend any other argument in F_s, $\Phi_{F_s}(\{a\}) = \{a\}$ an therefore $\mathit{grd}(F_s) = \{\{a\}\}$.

On the other hand consider AF F_u from Figure 2.3. All arguments have ingoing attacks, hence $\Phi_{F_u}(\emptyset) = \emptyset$ and $\mathit{grd}(F_u) = \{\emptyset\}$.

Note that, as the grounded semantics is unique, it holds that $\mathit{grd}(F) = \bigcap_{E \in \mathit{comp}(F)} E$ for each AF F. However the grounded extension does not coincide with the intersection of all preferred extensions in general. We show this in the following example.

Example 2.12. Consider the AF $F_g = (A, R)$ in Figure 2.4. We get $\mathit{pref}(F_g) = \{\{a, d\}, \{b, d\}\}$ and therefore $\bigcap_{E \in \mathit{pref}(F_g)} E = \{d\}$. But, as $\Phi_{F_g}(\emptyset) = \emptyset$, it follows that $\mathit{grd}(F_g) = \{\emptyset\}$.

We now turn to stable semantics. This semantics is based on stable models in logic programming [46], extensions in default logic [56], and stable expansions in autoepistemic logic [52]. The idea is that a stable extension must defeat each argument not in the extension, and by that let no argument "undecided"[2], i.e. an argument must either be in the extension or be attacked by an argument of the extension.

Definition 2.14. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *stable* extension in F if S is conflict-free in F and for all $a \in A \setminus S$ it holds that $S \mapsto_R a$. The collection of all stable extensions is denoted as $\mathit{stb}(F)$.

[1] As the grounded semantics is a unique status semantics, $\|\mathit{grd}(F)\| = 1$.

[2] In the labelling-based approach [25] the stable extensions are defined by enforcing that no argument has the label "undec".

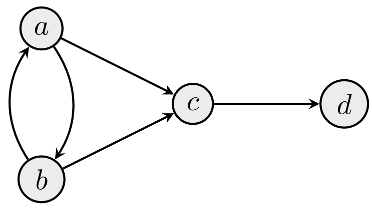

Figure 2.4: AF F_g with $\{\bigcap_{E \in \mathit{pref}(F_g)} E\} \neq \mathit{grd}(F_g)$ showing that $\{\bigcap_{E \in \mathit{pref}(F)} E\} = \mathit{grd}(F)$ does not hold in general

The range of a stable extension always covers all arguments of the AF, i.e. it holds that $S \in \mathit{stb}(F)$ only if $S_R^+ = A$.

As a stable extension is conflict-free and its range captures all arguments, the term conflict-free in Definition 2.14 can be replaced by one of the terms admissible, preferred and complete. Hence the following proposition follows immediately.

Proposition 2.8. *Given an arbitrary* AF $F = (A, R)$*, each stable extension is a preferred extension.*

The reverse does not hold. This becomes clear by the fact that there are argumentation frameworks which do not possess a stable extension, i.e. $\mathit{stb}(F) = \emptyset$. In contrast to all other semantics we deal with in this thesis, stable semantics are not "universally defined" [7].

Example 2.13. First take into account AF F_s from Figure 2.2. We get one stable extension, $\mathit{stb}(F_s) = \{a, d\}$. Consider, for example, the set $\{a, c\}$, then $\{a, c\}_R^+ = A_{F_s} \setminus \{e\}$, hence $\{a, c\}$ is no stable extension.

Now consider AF F_l from Figure 2.1. The only conflict-free sets are $\{b\}$ and $\{c\}$, and as their ranges do not encompass all arguments A_{F_s}, it follows that $\mathit{stb}(F_l) = \emptyset$, i.e. F_l has no stable extension.

The shortcomings of stable semantics are tackled by the following two semantics. We begin with the stage semantics. A stage extension is not required to attack all arguments which are not in the extension, but its range is required to be maximal with respect to set inclusion.

Definition 2.15. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *stage* extension in F if S is conflict-free in F and there is no $T \in \mathit{cf}(F)$ with $T_R^+ \supset S_R^+$. The collection of all stage extensions in F is denoted as $\mathit{stage}(F)$.

The following propositions show the strong connection between stable and stage semantics.

Proposition 2.9. *Given an arbitrary* AF $F = (A, R)$*, each stable extension in* F *is a stage extension in* F.

Proposition 2.10. *Given an arbitrary* AF $F = (A, R)$*, it holds that if there exists at least one stable extension in* F*, then each stage extension in* F *is a stable extension in* F.

It is easy to see that as for each AF there is at least one conflict-free set, there is also a range-maximal, conflict-free set, i.e. *stage*(F) is non-empty for each F.

All arguments not in the stage extension must be in some conflict with the extension, because if not they would have to be in the extension. From this fact the following proposition follows immediately:

Proposition 2.11. *Given an arbitrary* AF $F = (A, R)$*, each stage extension in* F *is a naive extension in* F.

Example 2.14. First consider the AF F_s from Figure 2.2. The set of stage extensions contains one element, *stage*$(F_s) = \{\{a, d\}\}$. Note that $\{a, d\}_R^+ = A_{F_s}$. Take, for example, the set $\{a, c\}$, then $\{a, c\}_R^+ = (A_{F_s} \setminus \{e\}) \subset A_{F_s}$, hence $\{a, c\}$ is no stage extension. So we have *stage*$(F_s) =$ *stb*(F_s). This shows that the reverse of Proposition 2.11 does not hold, as *naive*$(F_s) = \{\{a, c\}, \{a, d\}, \{b, d\}\}$ (cf. Example 2.8).

We have seen in Example 2.13 that for AF F_l from Figure 2.1 no stable extension exists. The only conflict-free sets are $\{b\}$ and $\{c\}$, with incomparable ranges $\{b\}_R^+ = \{a, b\}$ and $\{c\}_R^+ = \{b, c\}$. Therefore *stage*$(F_l) = \{\{b\}, \{c\}\}$.

The semi-stable semantics was mentioned in [60] as *admissible stages*, and extensively approached in [21]. Semi-stable semantics are the counterpart to preferred semantics as stage semantics are to naive semantics. The difference between semi-stable and preferred semantics is that with semi-stable semantics not only the extension itself, but its range is maximized.

Definition 2.16. Given an AF $F = (A, R)$, a set $S \subseteq A$ is a *semi-stable* extension in F if S is admissible in F and there is no $T \in$ *adm*(F) with $T_R^+ \supset S_R^+$. The collection of all semi-stable extensions in F is denoted as *sem*(F).

It is easy to see that as for each AF there is at least one admissible set, there is also a range-maximal, admissible set, i.e. *sem*(F) is non-empty for each F.

The following propositions dealing with the relationship of semi-stable semantics and stable semantics on the one hand and preferred semantics on the other hand are taken out of and proved in [21].

Proposition 2.12. *Given an arbitrary* AF $F = (A, R)$*, each stable extension in* F *is a semi-stable extension in* F.

Proposition 2.13. *Given an arbitrary* AF $F = (A, R)$*, each semi-stable extension in* F *is a preferred extension in* F.

Proposition 2.14. *Given an arbitrary* AF $F = (A, R)$*, it holds that if there exists at least one stable extension in* F*, then each semi-stable extension in* F *is a stable extension in* F.

Again we look at some examples:

Example 2.15. First consider AF F_s from Figure 2.2. We get one semi-stable extension, *sem*$(F_s) = \{\{a, d\}\}$. The range $\{a, d\}_R^+$ is A_{F_s} while all other admissible sets have smaller

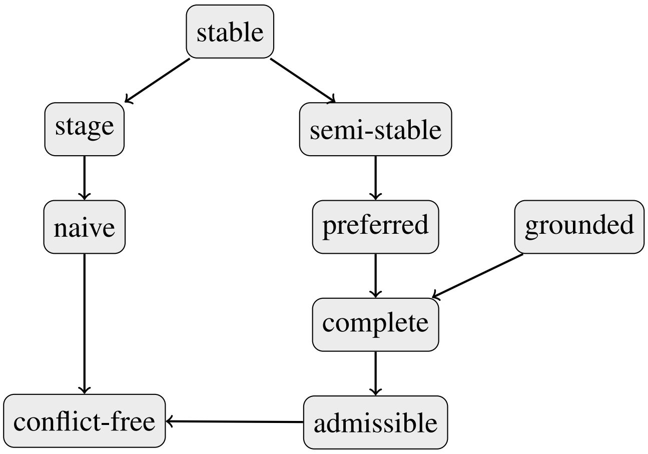

Figure 2.5: Relations between semantics

range. Note that $\{a, c\} \in \mathit{pref}(F_s)$ (cf. Example 2.9), but $\{a, c\} \notin \mathit{sem}(F_s)$, which shows that the reverse of Proposition 2.13 does not hold.

Now consider AF F_l from Figure 2.1. By Example 2.13 we know that $\mathit{stb}(F_l) = \emptyset$. As $\mathit{adm}(F_l) = \{\{c\}\}$ also $\mathit{sem}(F_l) = \{\{c\}\}$. Recall that $\mathit{stage}(F_l) = \{\{b\}, \{c\}\}$, which shows that semi-stable and stage semantics do not necessarily coincide if there is no stable extension.

The collection of "is-a"-relationships between the semantics covered in this chapter is summarized in Figure 2.5. An arrow from semantics σ_1 to semantics σ_2 means that each σ_1-extension is also a σ_2-extension. We have seen that reverse of none of these "is-a"-relations holds in general, i.e. for no semantics σ_1 and σ_2, where each σ_1-extension is also a σ_2-extension, it holds that each σ_2-extension is also a σ_1-extension in general. Therefore none of the semantics coincide. However recall that for all AFs where at least one stable extension exists, stage, semi-stable and stable semantics coincide.

Table 2.1 summarizes the examples of this chapter on the standard framework F_s from Example 2.2 and shows again which sets of arguments are extensions under the various semantics. A checkmark in row S and column σ has to be read as follows: S is an extension of F_s under the σ-semantics, i.e. $S \in \sigma(F)$. A "–" means that S is no extension of F_s under the σ-semantics, i.e. $S \notin \sigma(F)$.

In Chapter 1 we mentioned that identifying the set of valid arguments for the liar paradox is a non-trivial task. Table 2.2 shows the extensions for the AF F_l obtained from the liar paradox (see Figure 2.1) under the semantics introduced in this chapter. Indeed the majority of semantics (all except stable) agrees on c to be an extension. But naive and stage semantics, i.e. semantics not based on admissibility, also accept b as an extension. Further note that as all semantics introduced are based on conflict-freeness, no two- or more-element-extensions can be obtained from any of the semantics in this example.

An interesting subclass of AFs are *symmetric argumentation frameworks*. These are frameworks $F = (A, R)$ where the attack relation is symmetric, i.e. if $(a, b) \in R$ then also $(b, a) \in R$. They

	cf	*naive*	*stage*	*stb*	*adm*	*pref*	*sem*	*comp*	*grd*
$\emptyset$	✓	-	-	-	✓	-	-	-	-
$\{a\}$	✓	-	-	-	✓	-	-	✓	✓
$\{b\}$	✓	-	-	-	-	-	-	-	-
$\{c\}$	✓	-	-	-	✓	-	-	-	-
$\{d\}$	✓	-	-	-	✓	-	-	-	-
$\{e\}$	-	-	-	-	-	-	-	-	-
$\{a, c\}$	✓	✓	-	-	✓	✓	-	✓	-
$\{a, d\}$	✓	✓	✓	✓	✓	✓	✓	✓	-
$\{b, d\}$	✓	✓	-	-	-	-	-	-	-

Table 2.1: Extensions of F_s in Figure 2.2

were studied extensively in [29]. The authors make the following observations for symmetric AFs, where the attack relation is irreflexive, i.e. there are no self-attacking arguments.

Proposition 2.15. *For an irreflexive, symmetric* AF F*, the following holds:*

- $cf(F) = adm(F)$.
- $stb(F) = pref(F) = naive(F) = stage(F) = sem(F)$.
- $grd(F) = \{a \mid \nexists b : (b, a) \in R\}$.

As already mentioned in Section 2.1 we restrict ourselves to finite AFs. The reason for this is that in infinite AFs the notion of maximiality with respect to set inclusion is not as immediate as it is in finite frameworks. While the existence of complete, grounded and preferred extensions is guaranteed not only in the finite, but also in the infinite case (see [26, 33]), it was shown in [26, 60, 61] that there are infinite AFs where semi-stable and stage extensions do not exist.

	cf	*naive*	*stage*	*stb*	*adm*	*pref*	*sem*	*comp*	*grd*
$\emptyset$	✓	-	-	-	✓	-	-	-	-
$\{a\}$	-	-	-	-	-	-	-	-	-
$\{b\}$	✓	✓	✓	-	-	-	-	-	-
$\{c\}$	✓	✓	✓	-	✓	✓	✓	✓	✓

Table 2.2: Extensions of F_l in Figure 2.1

2.3 Propositional Logic

In Chapter 4, when we deal with realizing certain sets of extensions by constructing AFs we will make use of propositional logic. Parts of the construction process will be described by propositional formulas.

Therefore we give some basic definitions on propositional logic. We assume basic knowledge on syntax and semantics. For a more comprehensive introduction we refer to [43] or to [20] for the reader mastering the German language.

Definition 2.17. The *alphabet* of propositional logic is given by

- logical connectives $\vee$, $\wedge$ and $\neg$,
- a countable set of *propositional atoms* $P = \{a, b, c, \dots\}$,
- propositional constants $\top$ and $\bot$, and
- auxiliary symbols (and).

The alphabet is the set of symbols which can occur in a propositional formula. In order to actually build a formula, the symbols have to take on a certain structure. Therefor see the next definition.

Definition 2.18. A *propositional formula* over P is defined as follows:

1. Each propositional atom and constant is a formula.
2. If α and β are formulas, then also $(\neg\alpha)$, $(\alpha \vee \beta)$ and $(\alpha \wedge \beta)$ are formulas.
3. Formulas are solely given by 1. and 2.

Note that we may omit brackets in cases where no ambiguities arise.

In terms of semantics we use the standard techniques of two-valued propositional logic. We limit our introduction on the semantics of propositional logic to the following example.

Example 2.16. Consider the propositional formula $\mathcal{F}$ over the set of propositional atoms $P = \{a, b, c\}$,

$$\mathcal{F} = \neg(a \wedge (\neg(b \vee c))),$$

and take, for instance, the truth assignment $A = \{a \leftarrow \mathbf{true}, b \leftarrow \mathbf{false}, c \leftarrow \mathbf{false}\}$. As $b = \mathbf{false}$ and $c = \mathbf{false}$, $(b \vee c)$ evaluates to $\mathbf{false}$, but its negation evaluates to $\mathbf{true}$. Since $a = \mathbf{true}$, $(a \wedge (\neg(b \vee c)))$ evaluates to $\mathbf{true}$, which means that $\mathcal{F}$ evaluates to $\mathbf{false}$.

On the other hand consider a truth assignment A', where $a = \mathbf{false}$ and b and c have arbitrary truth values. With $a = \mathbf{false}$, $(a \wedge (\neg(b \vee c)))$ surely evaluates to $\mathbf{false}$, and therefore $\mathcal{F}$ is $\mathbf{true}$.

Definition 2.19. Given a propositional formula $\mathcal{F}$ over a set of propositional atoms P. Then an *interpretation* (over P) is a set $I \subseteq P$ with the intended meaning that each $p \in P$ is

- $\mathbf{true}$ if $p \in I$, and

- **false** if $p \notin I$.

I is a *model* of $\mathcal{F}$, denoted by $I \models \mathcal{F}$, if $\mathcal{F}$ evaluates to $\mathtt{true}$ under the truth-assignment provided by I.

The interpretation, given as a set of atoms, is an implicit assignment of truth-values.

Definition 2.20. Given two propositional formulas $\mathcal{F}_1$ and $\mathcal{F}_2$ over the same set of propositional atoms P. The formulas are *logically equivalent*, denoted by $\mathcal{F}_1 \equiv \mathcal{F}_2$, if the models of $\mathcal{F}_1$ coincide with the models of $\mathcal{F}_2$.

Example 2.17. Take into account the propositional formulas $\mathcal{F}_1 = a \wedge b \wedge c$ and $\mathcal{F}_2 = \neg((\neg a) \vee (\neg b) \vee (\neg c))$. Since for both formulas the only interpretation over $\{a, b, c\}$, which is also a model is $I = \{a, b, c\}$ it holds that $\mathcal{F}_1 \equiv \mathcal{F}_2$.

When describing the construction of an AF we will make use of normal forms.

Definition 2.21. A formula is in *disjunctive normal form (DNF)* if it is of the form

$$\bigvee_{i=1}^{n} (\bigwedge_{j=1}^{m(i)} a_{i,j})$$

where each $a_{i,j}$ is a *literal*, i.e. either an atom or the negation of an atom. A formula in DNF is *positive* if and only if no negation occurs in it. We call each conjunction of atoms $(a_{i,1} \wedge \cdots \wedge a_{i,m(i)})$ for each i a *term*.

Definition 2.22. A formula is in *conjunctive normal form (CNF)* if it is of the form

$$\bigwedge_{i=1}^{n} (\bigvee_{j=1}^{m(i)} a_{i,j})$$

where each $a_{i,j}$ is a literal. A formula in CNF is *positive* if and only if no negation occurs in it. We call each disjunction of atoms $(a_{i,1} \vee \cdots \vee a_{i,m(i)})$ for each i a *clause*. Further we may write a CNF-formula as a set of clauses

$$\{\gamma_1, \ldots, \gamma_n\}$$

where γ_i is just the set of literals in clause i for each i. The empty set of clauses corresponds to the formula $\top$.

Example 2.18. Consider the formula $\mathcal{D} = (a \wedge b) \vee (a \wedge c)$. $\mathcal{D}$ is in DNF. A logically equivalent formula in CNF is $\mathcal{C} = a \wedge (b \vee c)$, $\mathcal{D} \equiv \mathcal{C}$. The alternative notation of $\mathcal{C}$ in clause-form is $\{\{a\}, \{b, c\}\}$.

Proposition 2.16. *Given a formula $\mathcal{C} = \{\gamma_1, \ldots, \gamma_n\}$ in CNF, an interpretation I is a model of $\mathcal{C}$, denoted by $I \models \mathcal{C}$, if and only if I is a model of each clause, i.e. $I \models \gamma_i$ for each $i \in \{1, \ldots, n\}$.*

Example 2.19. Consider the CNF-formula $\mathcal{C} = \{\{a\}, \{b, c\}\}$ from Example 2.18. Each model M of $\mathcal{C}$ has to fulfill $M \models a$ and $M \models b \vee c$. Therefore the models over $\{a, b, c\}$ of $\mathcal{C}$ are $\{a, b\}$, $\{a, c\}$ and $\{a, b, c\}$. Note that these models of $\mathcal{C}$ are also models of the logically equivalent DNF-formula $\mathcal{D} = (a \wedge b) \vee (a \wedge c)$.

Proposition 2.17. *Each propositional formula (in DNF) can be converted to a logically equivalent formula in CNF. Each propositional formula (in CNF) can be converted to a logically equivalent formula in DNF.*

The process of conversion makes use of the following equivalences:

1. $a \vee (b \wedge c) \equiv (a \vee b) \wedge (a \vee c)$
2. $a \wedge (b \vee c) \equiv (a \wedge b) \vee (a \wedge c)$
3. $\neg(a \wedge b) \equiv (\neg a) \vee (\neg b)$
4. $\neg(a \vee b) \equiv (\neg a) \wedge (\neg b)$
5. $\neg\neg a \equiv a$

Equivalences 1 and 2 are called distributive laws, 3 and 4 form the De Morgan's laws and 5 can be referred to as double negative elimination.

As in the remainder of this thesis we will work with positive formulas in CNF (resp. DNF), the following observation will be useful.

Proposition 2.18. *Each positive DNF-formula can be converted to a logically equivalent positive CNF-formula. Each positive CNF-formula can be converted to a logically equivalent positive DNF-formula.*

The transformation process when dealing with positive CNF- or DNF-formulas only makes use of the distributive law. Nevertheless in terms of the size of the formula the conversion can in some cases lead to an exponential blow-up. For a formula in DNF containing n terms the logically equivalent formula in CNF can contain up to 2^n clauses. In [49] the authors investigate the size of CNF- and DNF-formulas in the context of conversion. We make this problem more concrete with an example.

Example 2.20. Consider the following formula $\mathcal{D}$ in DNF:

$$\mathcal{D} = (a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \cdots \vee (a_n \wedge b_n).$$

A minimal (in size) logically equivalent formula $\mathcal{C}$ in CNF is

$$
\begin{aligned}
\mathcal{C} = &(a_1 \vee a_2 \vee \cdots \vee a_{n-1} \vee a_n) \wedge \\
&(a_1 \vee a_2 \vee \cdots \vee a_{n-1} \vee b_n) \wedge \\
&(a_1 \vee a_2 \vee \cdots \vee b_{n-1} \vee a_n) \wedge \\
&(a_1 \vee a_2 \vee \cdots \vee b_{n-1} \vee b_n) \wedge \\
&\dots \\
&(b_1 \vee a_2 \vee \cdots \vee b_{n-1} \vee b_n) \wedge \\
&(a_1 \vee b_2 \vee \cdots \vee b_{n-1} \vee b_n) \wedge \\
&(b_1 \vee b_2 \vee \cdots \vee b_{n-1} \vee b_n).
\end{aligned}
$$

Each of the clauses contains either a_i or b_i for each $i = 1, \dots, n$. Therefore the number of clauses of $\mathcal{C}$ is exactly 2^n.

CHAPTER 3

Properties

Having introduced the basic concepts of abstract argumentation and propositional logic we are now ready to come to the first main part of the thesis. We know that the outcome of a semantics is a set of extensions. In this chapter we will introduce some properties such a set of extensions can fulfill or not. Further we will examine the relations between those properties, i.e. whether some properties imply another property. The second part of this chapter will be concerned with showing that given some semantics, the set of extensions of any argumentation framework fulfills certain properties. This gives us a first idea of how much disagreement a certain semantics can express. We will do this for every multiple status semantics introduced in Chapter 2. For the grounded semantics, a unique status semantics, there is not too much to say in this context, except of course that each extension-set under the grounded semantics contains exactly one extension, which also holds for every other unique status semantics.

3.1 Properties of Extension-Sets

When dealing with extension-based semantics in abstract argumentation, the result of applying a certain semantics to an AF is a set of extensions, i.e. a set of sets of arguments. We will use the term *extension-set* for those sets. This section gives some possible characteristics of extension-sets, which will be useful to describe the capabilities of the semantics.

We begin with some basic definitions.

Definition 3.1. Given a set $\mathcal{S} \subseteq 2^{\mathcal{A}}$. We use

- $Args_{\mathcal{S}}$ to denote $\bigcup_{S \in \mathcal{S}} S$ and
- $Pairs_{\mathcal{S}}$ to denote $\{(a, b) \mid \exists S \in \mathcal{S} : \{a, b\} \subseteq S\}$.

A set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ is called an *extension-set* (over $\mathcal{A}$) if $Args_{\mathcal{S}}$ is finite. We denote the set of all extension-sets over $\mathcal{A}$ as $\Sigma_{\mathcal{A}} = \{\mathcal{S} \subseteq 2^{\mathcal{A}} \mid Args_{\mathcal{S}} \text{ is finite}\}$.

In words, $\mathit{Args}_\mathcal{S}$ is the set of all arguments occurring in some element of an extension-set $\mathcal{S}$, while $\mathit{Pairs}_\mathcal{S}$ is the set of pairs of arguments going together in an extension $S \in \mathcal{S}$.

Example 3.1. We list some extension-sets, which we will use throughout this thesis. In this chapter will use them in order to check the presented properties on these sets.

- $\mathcal{S}_0 = \{\}$
- $\mathcal{S}_1 = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}\}$
- $\mathcal{S}_2 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}, \{b_1, b_2, b_3\}\}$
- $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$
- $\mathcal{S}_4 = \{\{a, b\}, \{a, c, e\}, \{b, d, e\}\}$
- $\mathcal{S}_5 = \{\{a\}, \{b, c\}, \{a, c, d\}\}$
- $\mathcal{S}_6 = \{\{a\}, \{b\}, \{c\}, \{a, b, c\}\}$
- $\mathcal{S}_7 = \{\{a, b\}, \{b, c\}, \{c, a\}\}$

Considering $\mathcal{S}_5$, we get

- $\mathit{Args}_{\mathcal{S}_5} = \{a, b, c, d\}$ and
- $\mathit{Pairs}_{\mathcal{S}_5} = \{(a, a), (a, c), (a, d), (b, b), (b, c), (c, c), (c, a), (c, b), (c, d), (d, d), (d, a), (d, c)\}$

Note that for any extension-set $\mathcal{S}$, it holds that $\mathit{Pairs}_\mathcal{S}$ is symmetric and reflexive, i.e.

- if $(a, b) \in \mathit{Pairs}_\mathcal{S}$ then $(b, a) \in \mathit{Pairs}_\mathcal{S}$, and
- for each $a \in \mathit{Args}_\mathcal{S}$, $(a, a) \in \mathit{Pairs}_\mathcal{S}$.

A simple characteristic of an extension-set is that for any element of the extension-set, all subsets of this element are in the extension-set too. We make this idea more formal with the following two definitions.

Definition 3.2. Given an extension-set $\mathcal{S} \subseteq 2^\mathcal{A}$, the *downward-closure* of $\mathcal{S}$, $\mathit{dcl}(\mathcal{S})$, is given by $\{S' \subseteq S \mid S \in \mathcal{S}\}$.

Definition 3.3. Given an extension-set $\mathcal{S} \subseteq 2^\mathcal{A}$, we call $\mathcal{S}$ *downward-closed* if $\mathcal{S} = \mathit{dcl}(\mathcal{S})$.

Example 3.2. It is easy to see that $\mathcal{S}_0$ and $\mathcal{S}_1$ are the only extension-sets given in Example 3.1, which are downward-closed. $\mathcal{S}_1$ contains all subsets of $\{a, c\}$, namely $\{a\}$, $\{c\}$ and $\emptyset$ as extensions. It is easy to see that $\mathcal{S}_2, \ldots, \mathcal{S}_7$ are not downward-closed, as none of them contains the empty set, which is contained in each downward-closure of a non-empty extension-set. As an example, the downward-closure of $\mathcal{S}_4$ is $\{\emptyset\} \cup \{\{s\} \mid s \in \mathit{Args}_{\mathcal{S}_4}\} \cup \{\{x, y\} \mid (x, y) \in \mathit{Pairs}_{\mathcal{S}_4}\} \cup \mathcal{S}_4$.

We are ready to introduce the next possible characteristic of an extension-set.

Definition 3.4. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *tight* if for all $S \in \mathcal{S}$ and $a \in \mathcal{A}$ it holds that if $(S \cup \{a\}) \notin \mathcal{S}$ then there exists an $s \in S$ such that $(a, s) \notin \mathit{Pairs}_{\mathcal{S}}$.

To describe the property of being tight, one could say that an argument a needs a reason for not going into an extension together with an element of the extension-set. Such a reason is, that it is not part of $\mathit{Pairs}_{\mathcal{S}}$ with at least one argument s of that element, i.e. there is no element of the extension-set, which is a superset of $\{a, s\}$.

The following example tests the extension-sets of Example 3.1 on being tight.

Example 3.3. One can observe that the extension-sets $\mathcal{S}_0$, $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$ are tight, while $\mathcal{S}_4$, $\mathcal{S}_5$, $\mathcal{S}_6$ and $\mathcal{S}_7$ are not. Consider $\mathcal{S}_3$ and take, for instance, $E = \{a_1, b_2, b_3\}$. Then none of (a_1, b_1), (a_1, a_2), and (a_1, a_3) is contained in $\mathit{Pairs}_{\mathcal{S}_3}$. The other two extensions of $\mathcal{S}_3$ behave in a symmetric way, meaning that $\mathcal{S}_3$ is tight.

On the other hand consider $\mathcal{S}_4$ and take $E' = \{a, b\}$. Then $(E' \cup \{e\}) \notin \mathcal{S}_4$, but both (a, e) and (b, e) are contained in $\mathit{Pairs}_{\mathcal{S}_4}$, showing that $\mathcal{S}_4$ is not tight.

Note that the extension-sets $\mathcal{S}_5$, $\mathcal{S}_6$ and $\mathcal{S}_7$ are not tight either.

The following property is actually the same as *I-maximality* in [5], but since at this point there is no reason to talk about maximality, we call it as what it prescribes, *incomparability*.

Definition 3.5. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *incomparable* if all elements $S \in \mathcal{S}$ are pairwise incomparable, i.e. for each $S, S' \in \mathcal{S}$ it holds that $S \subseteq S'$ implies $S = S'$.

Lemma 3.1. *For any incomparable extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is tight, then each $\mathcal{S}' \subseteq \mathcal{S}$ is tight too.*

Proof. Consider some tight extension-set $\mathcal{S}$ and a subset $\mathcal{S}' \subseteq \mathcal{S}$ thereof. From $\mathcal{S}' \subseteq \mathcal{S}$ we know that $\mathit{Pairs}_{\mathcal{S}'} \subseteq \mathit{Pairs}_{\mathcal{S}}$, $S \in \mathcal{S}' \Rightarrow S \in \mathcal{S}$ and with the fact that $\mathcal{S}$ is incomparable also $(S \cup \{a\}) \notin \mathcal{S}' \Rightarrow (S \cup \{a\}) \notin \mathcal{S}$ for any $a \in \mathcal{A}$. So if the implication of Definition 3.4 holds for all $S \in \mathcal{S}$, it also holds for all $S \in \mathcal{S}'$, which means that $\mathcal{S}'$ is tight. □

Corollary 3.2. *For any incomparable extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathit{dcl}(\mathcal{S})$ is tight, then $\mathcal{S}$ is tight too.*

The following example shows that for a tight extension-set $\mathcal{S}$ its downward-closure is not necessarily tight too. The reason for that is the following. Consider some extension-set $\mathcal{S}$, an element thereof $S \in \mathcal{S}$ and an argument a with $(S \cup \{a\}) \notin \mathcal{S}$. As $\mathcal{S}$ is tight there is some $s \in S$ with $(a, s) \notin \mathit{Pairs}_{\mathcal{S}}$. Assume $(a, t) \in \mathit{Pairs}_{\mathcal{S}}$ for all other $t \in S$. But now consider $S' = S \setminus \{s\}$. As $S' \in \mathit{dcl}(\mathcal{S})$ and $(a, t) \in \mathit{Pairs}_{\mathcal{S}}$ for all $t \in S'$, the downward-close of $\mathcal{S}$ is not tight. We make this more concrete with an example.

Example 3.4. To show that the other direction of Corollary 3.2 does not hold, i.e. that an extension-set being tight does not imply that its downward-closure is tight, consider $\mathcal{S}_3$ from Example 3.1. We have already seen in Example 3.3 that $\mathcal{S}_3$ is tight. However this is not the fact for its downward-closure $\mathit{dcl}(\mathcal{S}_3)$. In fact, $\{b_2, b_3\} \in \mathit{dcl}(\mathcal{S}_3)$ and now for b_1, we have that $\{b_1, b_2, b_3\} \notin dcl(\mathcal{S}_3)$ but both (b_1, b_2) and (b_1, b_3) are contained in $Pairs_{dcl(\mathcal{S}_3)} = Pairs_{\mathcal{S}_3}$. Therefore $dcl(\mathcal{S}_3)$ is not tight.

On the other hand, the downward-closure of $\mathcal{S}_2$, $dcl(\mathcal{S}_2)$, is tight, since $\{b_1, b_2, b_3\} \in \mathcal{S}_2$ and therefore of course $\{b_1, b_2, b_3\} \in dcl(\mathcal{S}_2)$.

The next property represents just another, though interesting, way of describing a downward-closed and tight extension-set.

Definition 3.6. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *cf-closed* if for all $A \subseteq Args_{\mathcal{S}}$ it holds that if $A \notin \mathcal{S}$, then there exist some $a, b \in A$ such that $(a, b) \notin Pairs_{\mathcal{S}}$.

Lemma 3.3. *For each extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that the following are equivalent:*

- *$\mathcal{S}$ is downward-closed and tight.*
- *$\mathcal{S}$ is cf-closed.*

Proof. In order to show the first direction consider a downward-closed and tight set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ which is not cf-closed. The latter means that there exists some $A \subseteq Args_{\mathcal{S}}$ such that $A \notin \mathcal{S}$ but $\forall a, b \in A$ it holds that $(a, b) \in Pairs_{\mathcal{S}}$. Consider some arbitrary $a, b \in A$. Since $\mathcal{S}$ is downward-closed we infer $\{a, b\} \in \mathcal{S}$ and further that there is some $B \in \mathcal{S}$ and some $c \in A$ such that $(B \cup \{c\}) \notin \mathcal{S}$ with $\{a, b\} \subseteq B \subset A$. This contradicts $\mathcal{S}$ being tight, because $(B \cup \{c\}) \notin \mathcal{S}$, but also $\forall s \in B : (s, c) \in Pairs_{\mathcal{S}}$.

To show the other direction consider a cf-closed set $\mathcal{S} \subseteq 2^{\mathcal{A}}$. First assume $\mathcal{S}$ is not downward-closed. Then there exist some A, B with $A \subset B \subseteq Args_{\mathcal{S}}$ such that $A \notin \mathcal{S}$ but $B \in \mathcal{S}$. Hence $\forall a, b \in B : (a, b) \in Pairs_{\mathcal{S}}$, and also $\forall a, b \in A : (a, b) \in Pairs_{\mathcal{S}}$, which contradicts to $\mathcal{S}$ being cf-closed. Finally assume $\mathcal{S}$ is not tight, i.e. there exists some $S \in \mathcal{S}$ and some $a \in \mathcal{A}$ such that $(S \cup \{a\}) \notin \mathcal{S}$, but $\forall s \in S : (a, s) \in Pairs_{\mathcal{S}}$. Since $S \in \mathcal{S}$ also $\forall b, c \in (S \cup \{a\}) : (b, c) \in Pairs_{\mathcal{S}}$, which, together with $(S \cup \{a\}) \notin \mathcal{S}$ contradicts $\mathcal{S}$ being cf-closed. □

Lemma 3.3 shows what was previously noted, namely that cf-closed is just a shortcut for downward-closed and tight. Computationally checking if an extension-set $\mathcal{S}$ is cf-closed can be very inefficient, as one has to check some property for each $A \subseteq Args_{\mathcal{S}}$. The definition of tight is more compact in that sense, as one only has to check the “outskirts” of each $S \in \mathcal{S}$. Therefore we will rather use the term tight in the remainder of the thesis.

We continue by introducing another property of extension-sets.

Definition 3.7. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *adm-closed* if for each $A, B \in \mathcal{S}$ the following holds: if $(a, b) \in Pairs_{\mathcal{S}}$ for each $a, b \in A \cup B$, then also $A \cup B \in \mathcal{S}$.

In words, for an extension-set to be adm-closed, if there is no good reason that the union of two elements of the extension-set is not an element of the extension-set, then it is. Such a good reason would be, if two arguments a and b of that union were not part of $Pairs_{\mathcal{S}}$, i.e. there is no element of the extension-set, which is a superset of $\{a, b\}$.

Again we look at an example.

Example 3.5. We again consider the extension-sets given in Example 3.1. Only $\mathcal{S}_6$ and $\mathcal{S}_7$ are not adm-closed, while $\mathcal{S}_0, \dots, \mathcal{S}_5$ are adm-closed. Consider the extension-set $\mathcal{S}_1$. Take, for instance, the extensions $\{a\}, \{c\} \in \mathcal{S}_1$, then the premise, namely that $(a, c), (c, a) \in \mathit{Pairs}_{\mathcal{S}_1}$, as well as the conclusion, that is $\{a, c\} \in \mathcal{S}_1$, hold. Since this is the only non-trivial case[1] where the premise of Definition 3.7 holds, $\mathcal{S}_1$ is adm-closed.

On the other hand take the extensions $\{a, b\}, \{b, c\} \in \mathcal{S}_7$. Then $(a, b), (b, c), (c, a)$ as well as their symmetric counterparts are in $\mathit{Pairs}_{\mathcal{S}_7}$. But $\{a, b, c\} \notin \mathcal{S}_7$, which means that $\mathcal{S}_7$ is not adm-closed.

We show another relation between properties:

Lemma 3.4. *For any extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is downward-closed and tight, then $\mathcal{S}$ is adm-closed.*

Proof. Consider an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ which is downward-closed and tight. To the contrary assume that $\mathcal{S}$ is not adm-closed, i.e. there are some $A, B \in \mathcal{S}$ with $(a, b) \in \mathit{Pairs}_{\mathcal{S}}$ for each $a, b \in A \cup B$, but $(A \cup B) \notin \mathcal{S}$. By Lemma 3.3 $\mathcal{S}$ is cf-closed, meaning that since $(A \cup B) \notin \mathcal{S}$ there exist some $a, b \in (A \cup B)$ such that $(a, b) \notin \mathit{Pairs}_{\mathcal{S}}$, a contradiction. □

Knowing about the property adm-closed we turn to a different, yet related property, as we will see in Lemma 3.8 and Lemma 3.9.

Definition 3.8. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *pref-closed* if for each $A, B \in \mathcal{S}$ with $A \neq B$, there exist $a, b \in (A \cup B)$ such that $(a, b) \notin \mathit{Pairs}_{\mathcal{S}}$.

So in a pref-closed extension-set $\mathcal{S}$, the union of any two elements of $\mathcal{S}$ must contain at least two arguments a and b such that (a, b) is not in $\mathit{Pairs}_{\mathcal{S}}$. The following lemma shows that this holds also for any incomparable and tight extension-set.

Lemma 3.5. *For any extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is incomparable and tight, then $\mathcal{S}$ is pref-closed.*

Proof. In order to show that this implication holds, consider some incomparable and tight extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ and assume towards a contradiction that $\mathcal{S}$ is not pref-closed. That means that there are some $A, B \in \mathcal{S}$ with $A \neq B$ such that for all $a, b \in (A \cup B)$ it holds that $(a, b) \in \mathit{Pairs}_{\mathcal{S}}$. Since $\mathcal{S}$ is incomparable, $B \neq \emptyset$ and $\forall b \in B : (A \cup \{b\}) \notin \mathcal{S}$. Considering an arbitrary $b \in B$ we get $\exists a \in A : (a, b) \notin \mathit{Pairs}_{\mathcal{S}}$ by the fact that $\mathcal{S}$ is tight. But this is a contradiction to $\forall a, b \in (A \cup B) : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$ and therefore the implication holds. □

The other direction does not hold. Particularly, a pref-closed extension-set is not necessarily tight. However, the following holds:

Lemma 3.6. *For any extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is pref-closed, then $\mathcal{S}$ is incomparable.*

[1] The trivial case is if $\emptyset$ is one of the two extensions; in this case the implication always holds.

Proof. Consider a pref-closed extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$. Towards a contradiction assume that $\mathcal{S}$ is not incomparable, i.e. $\exists A, B \in \mathcal{S} : A \subset B$. But then for all $a, b \in B : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$ and since $(A \cup B) = B$ also for all $a, b \in (A \cup B) : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$. Thus $\mathcal{S}$ is not pref-closed, a contradiction. □

In the following example we check, if the extension-sets given in Example 3.1 are pref-closed and show that the other direction of the implication of Lemma 3.5 does not hold, i.e. that a pref-closed extension-set is not tight in general[2].

Example 3.6. We show that the other direction of Lemma 3.5 does not hold. To this end consider $\mathcal{S}_4$ from Example 3.1, which is incomparable but not tight (see Example 3.3). Let $A = \{a, b\}$, $B = \{a, c, e\}$, $C = \{b, d, e\}$, and observe that $\mathcal{S}_4 = \{A, B, C\}$. $\mathcal{S}_4$ is pref-closed, because: $b, c \in (A \cup B)$ and $(b, c) \notin \mathit{Pairs}_{\mathcal{S}_4}$; $a, d \in (A \cup C)$ and $(a, d) \notin \mathit{Pairs}_{\mathcal{S}_4}$; $c, d \in (B \cup C)$ and $(c, d) \notin \mathit{Pairs}_{\mathcal{S}_4}$.

One can check that $\mathcal{S}_0, \ldots, \mathcal{S}_3$ are pref-closed (necessarily by Lemma 3.5). $\mathcal{S}_5$ is not pref-closed, which follows from the fact that $\mathcal{S}_5$ is not incomparable and Lemma 3.6. Also $\mathcal{S}_6$ and $\mathcal{S}_7$ are not pref-closed, in $\mathcal{S}_7$ we have, for instance, $\nexists x, y \in (\{a, b\} \cup \{b, c\})$ such that $(x, y) \notin \mathit{Pairs}_{\mathcal{S}_7}$.

A pref-closed extension-set also stays pref-closed when removing some of its elements:

Lemma 3.7. *Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ which is pref-closed, it holds that any $\mathcal{S}' \subseteq \mathcal{S}$ is pref-closed too.*

Proof. Consider some extension-sets $\mathcal{S}, \mathcal{S}'$ with $\mathcal{S}' \subseteq \mathcal{S}$ and, towards a contradiction, assume that $\mathcal{S}$ is pref-closed and $\mathcal{S}'$ is not. The latter means that $\exists A, B \in \mathcal{S}'$ with $A \neq B$, such that $\forall a, b \in (A \cup B) : (a, b) \in \mathit{Pairs}_{\mathcal{S}'}$. As $\mathcal{S}' \subseteq \mathcal{S}$ it holds that $A, B \in \mathcal{S}$ and $\mathit{Pairs}_{\mathcal{S}'} \subseteq \mathit{Pairs}_{\mathcal{S}}$. Therefore it holds that $\forall a, b \in (A \cup B) : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$, a contradiction to $\mathcal{S}$ being pref-closed. □

The properties adm-closed and pref-closed are closely related, as Lemma 3.8 and Lemma 3.9 show.

Lemma 3.8. *Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, it holds that $\mathcal{S}$ is pref-closed iff it is incomparable and adm-closed.*

Proof. We first show that $\mathcal{S}$ being incomparable and adm-closed implies that $\mathcal{S}$ is pref-closed. To this end let $\mathcal{S}$ be incomparable and adm-closed and $A, B \in \mathcal{S}$, with $A \neq B$. By incomparability, $A \neq B$ implies $(A \cup B) \notin \mathcal{S}$. Then since $\mathcal{S}$ is adm-closed, it follows that there exist $a, b \in (A \cup B)$ such that $(a, b) \notin \mathit{Pairs}_{\mathcal{S}}$. Hence $\mathcal{S}$ is pref-closed.

It remains to show that $\mathcal{S}$ being pref-closed implies that $\mathcal{S}$ is incomparable as well as adm-closed. We know from Lemma 3.6 that a pref-closed extension-set is incomparable. In order to show that a pref-closed extension-set is also adm-closed, consider an extension-set $\mathcal{S}$ which is not adm-closed. Then there are $A, B \in \mathcal{S}$ such that for all $a, b \in (A \cup B) : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$, which means that $\mathcal{S}$ is not pref-closed. □

[2] Each pref-closed extension-set is incomparable though. We showed this in Lemma 3.6.

Lemma 3.9. *For each incomparable extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, it holds that $\mathcal{S}$ is pref-closed iff $\mathcal{S} \cup \{\emptyset\}$ is adm-closed.*

Proof. By the fact that $Pairs_{\mathcal{S}} = Pairs_{\mathcal{S}\cup\{\emptyset\}}$ for all $\mathcal{S} \subseteq 2^{\mathcal{A}}$, following Definition 3.7 it holds that $\mathcal{S} \cup \emptyset$ is adm-closed iff $\mathcal{S}$ is adm-closed. Therefore both directions follow immediately from Lemma 3.8. □

Before introducing the next possible property of an extension-set we give the definition of *completion-sets*.

Definition 3.9. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ and two sets $A, B \in \mathcal{S}$, we define the completion-sets $\mathcal{C}_{\mathcal{S}}(A \cup B)$ of $A \cup B$ as the set of $\subseteq$-minimal sets $C \in \mathcal{S}$ with $C \supseteq (A \cup B)$. If $\mathcal{C}_{\mathcal{S}}(A \cup B)$ consists of exactly one such $\subseteq$-minimal set we denote this as $C_{\mathcal{S}}(A \cup B)$.

In words this means that for two extensions, the completion-sets are extensions that contain the union of these two extensions and are minimal with respect to set inclusion. There are cases where for two sets $A, B \in \mathcal{S}$ there is no completion-set, one unique completion-set or many completion-sets. Note that for $A \subset B$, $\mathcal{C}_{\mathcal{S}} = \{\{B\}\}$, i.e. $C_{\mathcal{S}} = \{B\}$. We illustrate the three cases by the following example.

Example 3.7. Consider the extension-set $\mathcal{S}_7$ from Example 3.1 and take, for instance, $A = \{a, b\}$ and $B = \{b, c\}$. Then there is no $C \in \mathcal{S}_7$ with $C \supseteq (A \cup B)$ and therefore $\mathcal{C}_{\mathcal{S}_7}(A \cup B) = \{\}$.

Next consider $\mathcal{S}_6$ and take $A = \{a\}$ and $B = \{b\}$. Then there is a unique completion-set and so $\mathcal{C}_{\mathcal{S}_6}(A \cup B) = \{\{a, b, c\}\}$, i.e. $C_{\mathcal{S}_6}(A \cup B) = \{a, b, c\}$.

Finally consider the extension-set $S^* = \{\{a\}, \{b\}, \{a, b, c\}, \{a, b, d\}\}$. For $A = \{a\}$ and $B = \{b\}$ we get two completion-sets, namely $\mathcal{C}_{S^*}(\{a, b\}) = \{\{a, b, c\}, \{a, b, d\}\}$.

Having at hand the concept of completion-sets, we can now give the definition of the property of an extension-set being comp-closed.

Definition 3.10. Given an extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$, we call $\mathcal{S}$ *comp-closed* if for each $A, B \in \mathcal{S}$ the following holds: if $(a, b) \in Pairs_{\mathcal{S}}$ for each $a, b \in (A \cup B)$, then there exists a unique completion-set $C \in \mathcal{C}_{\mathcal{S}}(A \cup B)$, i.e. $C_{\mathcal{S}}(A \cup B)$ is well-defined.

Lemma 3.10. *For each extension-set $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is adm-closed, then $\mathcal{S}$ is comp-closed.*

Proof. This follows immediately from the fact that given an extension-set $\mathcal{S}$ and $A, B \in \mathcal{S}$, $A \cup B \in \mathcal{S}$ implies $C_{\mathcal{S}}(A \cup B) = A \cup B$, i.e. $A \cup B$ is the unique-set in $\mathcal{C}_{\mathcal{S}}(A \cup B)$. □

One can see that comp-closed as a weakening of the notion adm-closed. For an extension-set to be adm-closed, the union of two elements has to be an element of the extension-set too, if there is no evidence of a conflict[3] between any pair of arguments of the union. For an extension-set to be comp-closed it is sufficient that any superset of the union (a unique one though) of these two elements is in the extension-set in this case.

We check the property of being comp-closed on the extension-sets of Example 3.1.

[3] In this case no evidence of a conflict means that no pair of arguments is in $Pairs_{\mathcal{S}}$.

	$\mathcal{S}_0$	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$	$\mathcal{S}_4$	$\mathcal{S}_5$	$\mathcal{S}_6$	$\mathcal{S}_7$
non-empty	-	✓	✓	✓	✓	✓	✓	✓
downward-closed	✓	✓	-	-	-	-	-	-
$dcl(\mathcal{S})$ tight	✓	✓	✓	-	-	-	-	-
tight	✓	✓	✓	✓	-	-	-	-
incomparable	✓	-	✓	✓	✓	-	-	✓
adm-closed	✓	✓	✓	✓	✓	✓	-	-
pref-closed	✓	-	✓	✓	✓	-	-	-
comp-closed	✓	✓	✓	✓	✓	✓	✓	-

Table 3.1: Properties of extension-sets

Example 3.8. We have already seen that $\mathcal{S}_0, \dots, \mathcal{S}_5$ are adm-closed. By Lemma 3.10 all these sets are also comp-closed.

Now consider $\mathcal{S}_6$ and take, for instance, $\{a\}, \{b\} \in \mathcal{S}_6$. Since $(a,a), (a,b), (b,a), (b,b) \in$ *Pairs*$_{\mathcal{S}_6}$, but $\{a,b\} \notin \mathcal{S}_6$, it is not adm-closed. But since $\{a,b,c\} \in \mathcal{S}_6$ and $\{a,b,c\} \supseteq \{a,b\}$, $\mathcal{C}_{\mathcal{S}_6}(\{a,b\}) = \{\{a,b,c\}\}$ and therefore $\mathcal{S}_6$ is indeed comp-closed.

On the other hand take the extensions $\{a,b\}, \{b,c\} \in \mathcal{S}_7$. Then $(a,b), (b,c), (c,a)$ as well as their symmetric counterparts are in *Pairs*$_{\mathcal{S}_7}$. As we have seen in Example 3.7, $\mathcal{C}_{\mathcal{S}_7}(\{a,b,c\}) = \emptyset$. This holds for any pairs of extensions in $\mathcal{S}_7$ symmetrically, which means that $\mathcal{S}_7$ is not comp-closed.

Table 3.1 summarizes the properties of the extension-sets presented in this section. We see that the examples confirm the relationships between the properties. All sets with a tight downward-closure are tight themselves, agreeing with Lemma 3.1 and Corollary 3.2. All downward-closed and tight sets are also adm-closed, according to Lemma 3.4. As suggested by Lemma 3.5, all incomparable and tight sets are pref-closed too. The extension-sets which are pref-closed coincide with those which are incomparable and adm-closed, in accordance with Lemma 3.8. Finally, all adm-closed extension-sets are also comp-closed, backing up Lemma 3.10.

3.2 Properties of Argumentation Semantics

In Section 3.1 we defined some properties for extension-sets and examined the relations between these properties. Now we are going to go through all semantics introduced in Section 2.2, one after the other, and check, which properties generally hold for an extension-set of an arbitrary AF under a given semantics.

Properties of *cf*-based Semantics

We begin with semantics, whose definition is mainly based on the concept of conflict-freeness.

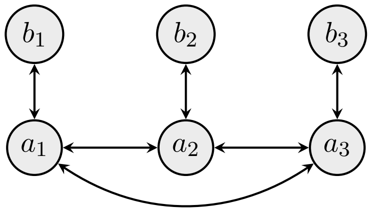

Figure 3.1: Argumentation framework F_n with $naive(F_n) = \mathcal{S}_2$

Conflict-free sets

Proposition 3.11. *For each* AF $F = (A, R)$ *it holds that* $cf(F)$ *is a non-empty, downward-closed and tight extension-set.*

Proof. Since $\emptyset$ is always conflict-free, $cf(F)$ is non-empty.

For any conflict-free set S it holds that all its subsets are conflict-free too, which means that $cf(F)$ is downward-closed.

In order to show that $cf(F)$ is tight let $S \in cf(F)$ and $a \in A$, such that $(S \cup \{a\}) \notin cf(F)$. It follows that there exists an argument $s \in S$ such that $s \mapsto a$ or $a \mapsto s$. Then $\{a, s\} \notin cf(F)$ and since $cf(F)$ is downward-closed, $\{a, s\} \not\subseteq T$ for any $T \in cf(F)$. It follows that $(a, s) \notin Pairs_{cf(F)}$ and therefore the implication of Definition 3.4 holds. □

We can also give an alternative characterization.

Corollary 3.12. *For each* AF $F = (A, R)$ *it holds that* $cf(F)$ *is a non-empty and cf-closed extension-set.*

Proof. This follows immediately from Proposition 3.11 and Lemma 3.3, which says that each downward-closed and tight extension-set is also cf-closed. □

Naive semantics

Proposition 3.13. *For each* AF $F = (A, R)$*,* $naive(F)$ *is a non-empty and incomparable extension-set where its downward-closure* $dcl(naive(F))$ *is tight.*

Proof. Since the naive extensions are the maximal conflict-free sets in terms of set-inclusion, $naive(F)$ is non-empty ($cf(F)$ is non-empty too) and incomparable.

From subset-maximality we observe that $dcl(naive(F)) = cf(F)$. Since $cf(F)$ is tight we conclude that $dcl(naive(F))$ is tight. □

A set of naive extensions is not only tight, but also its downward-closure is tight. For the AF F_n in Figure 3.1 it holds that $naive(F_n) = \mathcal{S}_2$ ($\mathcal{S}_2$ was defined in Example 3.1). This is in accordance with Proposition 3.13 as $\mathcal{S}_2$ is of course non-empty and incomparable and the fact that its downward-closure is tight was discussed in Example 3.4. Further note that the direction of the attacks in F_n does not affect the outcome of the naive semantics. Also for the AF F_{st}, which is depicted in Figure 3.2, it holds that $naive(F_{st}) = \mathcal{S}_2$.

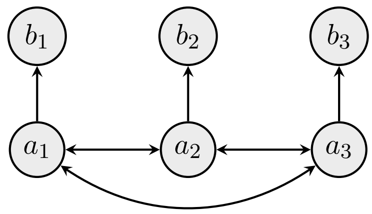

Figure 3.2: Argumentation framework F_{st} with $stage(F_{st}) = stb(F_{st}) = \mathcal{S}_3$

Stage semantics

Proposition 3.14. *For each* AF $F = (A, R)$, $stage(F)$ *is a non-empty, incomparable and tight extension-set.*

Proof. As stage extensions are range maximal conflict-free sets and there is always at least one conflict-free set, $stage(F)$ is non-empty.

We know from Proposition 2.11 that $stage(F) \subseteq naive(F)$. Since $naive(F)$ is incomparable and every subset of an incomparable set is incomparable itself, $stage(F)$ is incomparable.

Recall from Proposition 3.13 that $dcl(naive(F))$ is tight. Knowing that $naive(F)$ is incomparable we infer by Corollary 3.2 that $naive(F)$ is tight. Since $stage(F) \subseteq naive(F)$ it remains to show that for each incomparable $\mathcal{S} \subseteq 2^{\mathcal{A}}$ it holds that if $\mathcal{S}$ is tight, then $\mathcal{S}'$ is tight for each $\mathcal{S}' \subseteq \mathcal{S}$. Let $\mathcal{S}, \mathcal{S}' \subseteq 2^{\mathcal{A}}$ with $\mathcal{S}' \subseteq \mathcal{S}$ and let $\mathcal{S}$ be tight. Since $Pairs_{\mathcal{S}'} \subseteq Pairs_{\mathcal{S}}$ and $(S \cup \{a\}) \notin \mathcal{S}' \Rightarrow (S \cup \{a\}) \notin \mathcal{S}$, the implication of Definition 3.4 holds for all $S \in \mathcal{S}'$ and $a \in A$, and therefore $\mathcal{S}'$ is tight. □

Proposition 3.14 gives weaker properties for extension-sets under the stage semantics than Proposition 3.13 does for extension-sets under the naive semantics. While, given an arbitrary AF F, $dcl(naive(F))$ is always tight, for stage semantics only $stage(F)$ is always tight. The AF F_{st} in Figure 3.2 shows that $dcl(stage(F_{st}))$ is not tight as $stage(F_{st}) = \mathcal{S}_3$ ($\mathcal{S}_3$ is taken from Example 3.1) and the downward-closure of $\mathcal{S}_3$ is not tight by Example 3.4. Therefore $dcl(stage(F))$ is not necessarily tight, given some framework F.

Stable semantics We now turn to stable semantics. We treat the stable semantics as *cf*-based semantics, although of course each stable extension is an admissible set (see the summary of "is-a"-relations in Figure 2.5), since the stable semantics implicitly follows the concept of admissibility. The reasons for categorizing stable semantics as cf-based are on the one hand that the basic definition of stable semantics only uses conflict-freeness and not admissibility, and on the other hand that when it comes to realizing certain extension-sets under the stable semantics (we will come to that in Section 4.1), the construction process will have much more in common with the naive and stage semantics than with semantics which are admissibility-based in the narrower sense of the word, as admissible, preferred, semi-stable and complete semantics.

Proposition 3.15. *For each* AF $F = (A, R)$, $stb(F)$ *is an incomparable and tight extension-set.*

Proof. First note that if *stb*(F) is the empty set of extensions, the proposition holds, since $\emptyset$ is incomparable and tight by definition.

If *stb*$(F) \neq \emptyset$ we can, knowing that *stb*$(F) \subseteq$ *stage*(F), use the same arguments as in the proof of Proposition 3.14, i.e. every subset of an incomparable and tight set is an incomparable and tight set itself. What follows is that *stb*(F) is incomparable and tight. □

Properties of *adm*-based Semantics

We now turn to semantics, which are based on the central concept of admissibility.

Admissible sets

Proposition 3.16. *For each* AF $F = (A, R)$*, adm*(F) *is an adm-closed extension-set containing* $\emptyset$*.*

Proof. By definition $\emptyset$ is always admissible. We show that *adm*(F) is adm-closed. Towards a contradiction, assume that *adm*(F) is not adm-closed, i.e. there exist some $B, C \in$ *adm*(F) such that for all $b, c \in (B \cup C)$ it holds that $(b, c) \in \mathit{Pairs}_{\mathit{adm}(F)}$, but $(B \cup C) \notin$ *adm*(F). From Lemma 2.1 we know that since both B and C are admissible in F, $B \cup C$ is defended by itself in F. So for $(B \cup C) \notin$ *adm*(F) there must be a conflict in $B \cup C$, i.e. $\exists(b, c) \in R$ such that $\{b, c\} \subseteq (B \cup C)$. But then, for all $D \in$ *adm*(F), $\{b, c\} \nsubseteq D$. Hence by Definition 3.1, $(b, c) \notin \mathit{Pairs}_{\mathit{adm}(F)}$, a contradiction. □

By containing $\emptyset$, *adm*(F) is also non-empty for each F.

Preferred semantics

Proposition 3.17. *For each* AF $F = (A, R)$*, pref*(F) *is a non-empty and pref-closed extension-set.*

Proof. By definition the preferred semantics always proposes at least one extension, i.e. *pref*(F) is non-empty.

To show that *pref*(F) is a pref-closed extension-set consider two extensions $B, C \in$ *pref*(F) with $B \neq C$. Towards a contradiction let us assume that for all $a, b \in (B \cup C)$ it holds that $(a, b) \in \mathit{Pairs}_{\mathit{pref}(F)}$. Hence, $(B \cup C) \in$ *cf*(F) and by Lemma 2.1 all arguments in $B \cup C$ are defended by $B \cup C$. Thus $(B \cup C) \in$ *adm*(F). But now $B \cup C$ is an admissible superset of the preferred extensions B and C, a contradiction to the subset-maximality of preferred extensions. □

Proposition 3.17 suggests that the variety of extension-sets under the preferred semantics is bigger than the one of extension-set under the stage and stable semantics. While, given an arbitrary AF, the set of stage and stable extensions is tight, the set of preferred extensions is only pref-closed, which is weaker than tight (cf. Proposition 3.5 and Example 3.6). Figure 3.3 shows the AF F_p. The extension-set $\mathcal{S}_4$ from Example 3.1 coincides with the preferred extensions of F_p, *pref*(F_p). As $\mathcal{S}_4$ is not tight, there cannot be any framework F with *stage*$(F) = \mathcal{S}_4$ or *stb*$(F) = \mathcal{S}_4$.

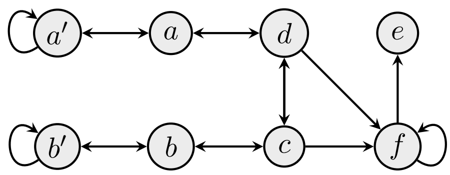

Figure 3.3: Argumentation framework F_p with *pref*$(F_p) = \mathcal{S}_4$

We can also give an alternative characterization of the set of preferred extensions.

Corollary 3.18. *For each* AF $F = (A, R)$*, pref*(F) *is a non-empty, incomparable and adm-closed extension-set.*

Proof. This follows immediately from Proposition 3.17 and Lemma 3.8, which says that each extension-set which is pref-closed is also adm-closed and incomparable. □

Semi-stable semantics Turning to semi-stable semantics we will see that a set of extensions under the semi-stable semantics fulfills the same properties as an extension-set under the preferred semantics (cf. Proposition 3.17).

Proposition 3.19. *For each* AF $F = (A, R)$*, sem*(F) *is a non-empty and pref-closed extension-set.*

Proof. By definition the semi-stable semantics always proposes at least one extension, which means that *sem*(F) is non-empty. By the facts that *sem*$(F) \subseteq$ *pref*(F) (see Proposition 2.13) and that *pref*(F) is pref-closed (see Proposition 3.17), we get that *sem*(F) is pref-closed by Lemma 3.7. □

Complete semantics

Proposition 3.20. *For each* AF $F = (A, R)$*, comp*(F) *is a non-empty, comp-closed extension-set where* $\bigcap_{S \in \mathit{comp}(F)} S \in \mathit{comp}(F)$*.*

Proof. First note that there is always at least one complete extension, namely the grounded extension. Moreover the grounded extension is the unique $\subseteq$-minimal complete extensions and hence $\bigcap_{S \in \mathit{comp}(F)} S \in \mathit{comp}(F)$. Finally consider two complete extensions E, E' such that $(a, b) \in \mathit{Pairs}_{\mathit{comp}(F)}$ for each $a, b \in (E \cup E')$. By Lemma 2.1, $E \cup E'$ is an admissible set and thus can be extended to a unique complete extension $C \supseteq (E \cup E')$ by Lemma 2.6. Therefore *comp*(F) is comp-closed. □

Proposition 3.20 suggests the weakest restrictions for extension-sets under the complete semantics compared to all semantics considered. We will see in Section 4.1 that the set of extension-sets where one can find a corresponding AF is largest for the complete semantics compared to the other semantics too.

	cf	*naive*	*stage*	*stb*	*adm*	*pref*	*sem*	*comp*
non-empty	✓	✓	✓	-	✓	✓	✓	✓
downward-closed	✓	-	-	-	-	-	-	-
$dcl(\mathcal{S})$ tight	✓	✓	-	-	-	-	-	-
tight	✓	✓	✓	✓	-	-	-	-
incomparable	-	✓	✓	✓	-	✓	✓	-
adm-closed	✓	✓	✓	✓	✓	✓	✓	-
pref-closed	-	✓	✓	✓	-	✓	✓	-
comp-closed	✓	✓	✓	✓	✓	✓	✓	✓

Table 3.2: Properties of semantics

Summarizing the results of this chapter we have seen that necessary properties of extension-sets differ between the semantics with the exception of semi-stable and preferred semantics. We will see in Section 4.1 that these necessary properties are also sufficient properties in order to realize an extension-set by an AF, i.e. to find an AF which provides the extension-set under the particular semantics.

Table 3.2 gives an overview of the necessary properties of extension-sets under the different semantics. A checkmark in line p and column σ has to be read as follows: For each AF F the set of extensions under semantics σ, $\sigma(F)$, fulfills property p.

CHAPTER 4

Realizability

In Chapter 3 we have introduced several properties of extension-sets and showed that extension-sets under certain semantics always fulfill certain properties. This chapter aims at finding properties of extension-sets such that, given a semantics, one can always find an AF where the extensions under the semantics coincide with the given extension-set, i.e. the extension-set is realizable under the semantics. We will first look at *general realizability*, where we do not restrict the number of arguments of the AF. Second we will introduce *strict realizability* by only allowing arguments which occur in the given extension-set to be in the set of arguments of the constructed AF. We will see that the characteristics for general and strict realizability do not coincide in general.

4.1 General Realizability

In the previous chapter we have given necessary characteristics for the extension-sets under the semantics σ, where $\sigma \in \{\mathit{cf}, \mathit{adm}, \mathit{naive}, \mathit{stb}, \mathit{stage}, \mathit{pref}, \mathit{sem}, \mathit{comp}\}$ are the semantics of our interest. Now we will show that these characteristics are also sufficient. So we investigate which sets of extensions can be realized by argumentation frameworks under a particular semantics σ. To this end, we require the concept of *realizability*.

Definition 4.1. Given a semantics σ, an extension-set $\mathcal{S}$ is called *σ-realizable* if there exists an AF F such that $\sigma(F) = \mathcal{S}$. $\mathcal{S}$ is then *realized* by F under σ.

This turns our characteristics into the desired characterizations for the signatures Σ_σ, which we will formally introduce in Chapter 5.

In order to show realizability, we will define several canonical argumentation frameworks, where each of these frameworks will be based on a given extension-set. If the extensions (under a semantics σ) of this AF coincide with the ones of the extension-set it was built upon for every extension-set (fulfilling certain properties), realizability is shown. So the definitions of these canonical frameworks can be seen as construction-guidelines of AFs realizing the given extension-set under a semantics σ.

Realizability of *cf*-based Semantics

We start with the following concept of a canonical argumentation framework, which will underlie all subsequent results on realizability.

Construction 4.1. Given an extension-set $\mathcal{S}$, we define the canonical argumentation framework $F^{cf}_{\mathcal{S}}$ as

$$F^{cf}_{\mathcal{S}} = \big(\mathit{Args}_{\mathcal{S}}, (\mathit{Args}_{\mathcal{S}} \times \mathit{Args}_{\mathcal{S}}) \setminus \mathit{Pairs}_{\mathcal{S}}\big).$$

$F^{cf}_{\mathcal{S}}$ is a symmetric argumentation framework, i.e. it only consists of symmetric attacks. Also observe that $F^{cf}_{\mathcal{S}}$ contains no self-attacking arguments since $(a, a) \in \mathit{Pairs}_{\mathcal{S}}$ and therefore $(a, a) \notin R_{F^{cf}_{\mathcal{S}}}$ for each $a \in \mathit{Args}_{\mathcal{S}}$. The underlying idea for specifying the relation is simple. Whenever two arguments occur jointly in a set $S \in \mathcal{S}$, we must not draw a relation between these two arguments; otherwise we do so. The fact that two arguments a and b occur jointly in a set $S \in \mathcal{S}$ is indicated by $(a, b) \in \mathit{Pairs}_{\mathcal{S}}$ (cf. Definition 3.1). Also note that the size of $F^{cf}_{\mathcal{S}}$ is polynomial in the size of $\mathcal{S}$.

Of course this is just a basic construction and will not suffice to cover realizability of all semantics. For $\mathcal{S}_2 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}, \{b_1, b_2, b_3\}\}$ (see Example 3.1), $F^{cf}_{\mathcal{S}_2}$ is exactly the AF depicted in Figure 3.1. As easily checked, $\mathit{naive}(F^{cf}_{\mathcal{S}_2}) = \mathcal{S}_2$ holds. When we consider the extension-set $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$, i.e. $\mathcal{S}_3 = \mathcal{S}_2 \setminus \{\{b_1, b_2, b_3\}\}$, we obtain the same framework $F^{cf}_{\mathcal{S}_3} = F^{cf}_{\mathcal{S}_2}$. In terms of naive semantics, this is not problematic since there cannot be an AF F, where the extensions under the naive semantics coincide with $\mathcal{S}_3$, since the downward-closure of $\mathcal{S}_3$ is not tight. We have already discussed this in Section 3.2. However, this observation readily suggests that realizing $\mathcal{S}_3$ with, say, stable semantics, requires additional concepts. We will come back to this issue later in this chapter[1], but first state some formal results on the canonical framework as defined in Construction 4.1.

Proposition 4.1. *For each non-empty, downward-closed, and tight extension-set $\mathcal{S}$, it holds that $\mathit{cf}(F^{cf}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. To show that $\mathit{cf}(F^{cf}_{\mathcal{S}}) \subseteq \mathcal{S}$, observe that for each $E \in \mathit{cf}(F^{cf}_{\mathcal{S}})$, there are no $a, b \in E$ with $(a, b) \in R_{F^{cf}_{\mathcal{S}}}$. Therefore by construction of $R_{F^{cf}_{\mathcal{S}}}$, it holds that $(a, b) \in \mathit{Pairs}_{\mathcal{S}}$ for all $a, b \in E$. Now suppose there exists an $E' \in \mathit{cf}(F^{cf}_{\mathcal{S}})$ such that $E' \notin \mathcal{S}$. Without loss of generality let E' be $\subseteq$-minimal with this property. Then $E' = (S \cup \{c\})$ for some $S \in \mathcal{S}$. As $\mathcal{S}$ is tight there is an $s \in S$ such that $(s, c) \notin \mathit{Pairs}_{\mathcal{S}}$, a contradiction to the above observation.

To show that $\mathit{cf}(F^{cf}_{\mathcal{S}}) \supseteq \mathcal{S}$, consider some $S \in \mathcal{S}$. Then all $a, b \in S$ are contained as pairs $(a, b) \in \mathit{Pairs}_{F^{cf}_{\mathcal{S}}}$, thus by construction of $F^{cf}_{\mathcal{S}}$, it holds that $(a, b) \notin R_{F^{cf}_{\mathcal{S}}}$ for all $a, b \in S$. Hence $S \in \mathit{cf}(F^{cf}_{\mathcal{S}})$. $\square$

Example 4.1. Figure 4.1 shows $F^{cf}_{\mathcal{S}_1}$, which is obtained from Construction 4.1 on the basis of $\mathcal{S}_1 = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}\}$ from Example 3.1. $F^{cf}_{\mathcal{S}_1}$ does not contain an attack between a

[1] Construction 4.2 will introduce the additional concepts needed for stable and stage semantics.

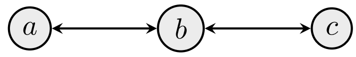

Figure 4.1: AF $F^{cf}_{\mathcal{S}_1}$ with $cf(F^{cf}_{\mathcal{S}_1}) = \mathcal{S}_1$ where $\mathcal{S}_1 = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}\}$

and c since $(a, c) \in \mathit{Pairs}_{\mathcal{S}_1}$. It is easy to verify that $F^{cf}_{\mathcal{S}_1}$ realizes $\mathcal{S}_1$ under the conflict-free semantics, i.e. $cf(F^{cf}_{\mathcal{S}_1}) = \mathcal{S}_1$.

We turn to naive semantics and approach the characterization for naive-realizable sets by the following result, which will be useful later.

Lemma 4.2. *For each incomparable and tight extension-set $\mathcal{S}$, it holds that $\mathcal{S} \subseteq naive(F^{cf}_{\mathcal{S}})$.*

Proof. Towards a contradiction assume there is some $S \in \mathcal{S}$ such that $S \notin naive(F^{cf}_{\mathcal{S}})$. Then either $S \notin cf(F^{cf}_{\mathcal{S}})$ or $\exists S' \supset S : S' \in cf(F^{cf}_{\mathcal{S}})$. If $S \notin cf(F^{cf}_{\mathcal{S}})$, then by Construction 4.1 $\exists a, b \in S : (a, b) \notin \mathit{Pairs}_{\mathcal{S}}$, a contradiction to $S \in \mathcal{S}$. Thus $\exists S' \supset S : S' \in cf(F^{cf}_{\mathcal{S}})$. Then by construction of $F^{cf}_{\mathcal{S}}$ it holds that $\forall a, b \in S' : (a, b) \in \mathit{Pairs}_{\mathcal{S}}$. Now consider some $a \in S' \setminus S$. Since $\mathcal{S}$ is incomparable, $S \cup \{a\} \notin \mathcal{S}$. Therefore, since $\mathcal{S}$ is tight, $\exists b \in S : (a, b) \notin \mathit{Pairs}_{\mathcal{S}}$, a contradiction. □

So when constructing $F^{cf}_{\mathcal{S}}$ for an incomparable and tight set $\mathcal{S}$, every $S \in \mathcal{S}$ is a naive extension of $F^{cf}_{\mathcal{S}}$. To ensure that the framework does not give any further naive extensions, the necessary properties of $\mathcal{S}$ have to be even stricter:

Proposition 4.3. *For each incomparable and non-empty extension-set $\mathcal{S}$, where $dcl(\mathcal{S})$ is tight, it holds that $naive(F^{cf}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. Consider some incomparable and non-empty extension-set $\mathcal{S}$, where $dcl(\mathcal{S})$ is tight. Since $dcl(\mathcal{S})$ is surely downward-closed, as well as tight and non-empty by the assumption, we know from Proposition 4.1 that $cf(F^{cf}_{dcl(\mathcal{S})}) = dcl(\mathcal{S})$. By $\mathit{Args}_{dcl(\mathcal{S})} = \mathit{Args}_{\mathcal{S}}$ and $\mathit{Pairs}_{dcl(\mathcal{S})} = \mathit{Pairs}_{\mathcal{S}}$ also $F^{cf}_{dcl(\mathcal{S})} = F^{cf}_{\mathcal{S}}$ holds, hence $cf(F^{cf}_{\mathcal{S}}) = dcl(\mathcal{S})$. By construction of $dcl(\mathcal{S})$ the $\subseteq$-maximal sets in $dcl(\mathcal{S})$ are the sets $S \in \mathcal{S}$ (note that $\mathcal{S}$ is incomparable by assumption) and as naive sets are just $\subseteq$-maximal conflict-free, $naive(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. □

So for the set of naive extensions of $F^{cf}_{\mathcal{S}}$ to be equal to the extension-set $\mathcal{S}$ it was built upon, not only $\mathcal{S}$ has to be tight but also its downward-closure $dcl(\mathcal{S})$.

Example 4.2. As already stated before, the AF in Figure 3.1 is just $F^{cf}_{\mathcal{S}_2}$, which we get by Construction 4.1 on the extension-set $\mathcal{S}_2$, for which it holds that it is non-empty, incomparable and its downward-closure $dcl(\mathcal{S}_2)$ is tight. As easily checked, $naive(F^{cf}_{\mathcal{S}_2}) = \mathcal{S}_2$. We get the same AF $F^{cf}_{\mathcal{S}_3} = F^{cf}_{\mathcal{S}_2}$ for the tight set $\mathcal{S}_3$. However $dcl(\mathcal{S}_3)$ is not tight. Confirming Lemma 4.2 and Proposition 4.3 we get that $naive(F^{cf}_{\mathcal{S}_3}) = \mathcal{S}_2$ and therefore $\mathcal{S}_3 \subseteq naive(F^{cf}_{\mathcal{S}_3})$, but $\mathcal{S}_3 \neq naive(F^{cf}_{\mathcal{S}_3})$.

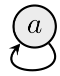

Figure 4.2: F_{prim} having $stb(F_{prim}) = \emptyset$

So far, in order to realize a set $\mathcal{S}$ we used a framework from $AF_{\mathcal{A}}$ of the form (A, R) with $A = Args_{\mathcal{S}}$, i.e. no additional arguments have to be introduced in order to construct the framework. For the subsequent results we require, in general, frameworks with $A \supset Args_{\mathcal{S}}$. In Section 4.2 we will go into more detail on this issue and show that this cannot be avoided. For the moment, we recall that $\mathcal{A}$ is infinite, hence there are always enough arguments available in $\mathcal{A}$.

Let us proceed with stable and stage semantics. Stable semantics are the only semantics that can realize $\mathcal{S} = \emptyset$. Note that $\mathcal{S} = \emptyset$ is easily *stb*-realizable:

Example 4.3. The primitive framework F_{prim} in Figure 4.2 has no stable extensions, therefore $stb(F_{prim}) = \emptyset$. All other semantics σ we deal with in this thesis have the empty set of arguments as single extension, $\sigma(F_{prim}) = \{\emptyset\}$. To get the empty set of arguments as single extension with stable semantics an empty AF $F_{empty} = (\emptyset, \emptyset)$ is needed. It holds that $stb(F_{empty}) = \{\emptyset\}$ as well as $\sigma(F_{empty}) = \{\emptyset\}$ for all other semantics σ.

If we look back at Proposition 3.14 and Proposition 3.15 the only difference between the possible sets of extensions one can get when dealing with stable and stage semantics was the case $\mathcal{S} = \emptyset$. The next results show that this indeed is the only difference for stable and stage semantics when it comes to realizing extension-sets.

Before we can do this, we need a more involved canonical framework. As the canonical framework from Construction 4.1 is symmetric, it holds that for each extension-set $\mathcal{S}$, $naive(F^{cf}_{\mathcal{S}}) = stb(F^{cf}_{\mathcal{S}}) = stage(F^{cf}_{\mathcal{S}})$ (cf. Proposition 2.15). So also with stable and stage semantics we have the problem of undesired extensions for canonical frameworks built upon extension-sets with a non-tight downward-closure. But in contrast to naive semantics we can overcome this problem with the following construction, which is inspired by a translation in [38].

Construction 4.2. Given an extension-set $\mathcal{S}$, we define the canonical argumentation framework $F^{st}_{\mathcal{S}}$ as

$$F^{st}_{\mathcal{S}} = \left(A^{st}_{\mathcal{S}}, R^{st}_{\mathcal{S}}\right),$$

with

$$\begin{aligned} A^{st}_{\mathcal{S}} &= Args_{\mathcal{S}} \cup \{\bar{E} \mid E \in \mathcal{X}\}, \text{ and} \\ R^{st}_{\mathcal{S}} &= ((Args_{\mathcal{S}} \times Args_{\mathcal{S}}) \setminus Pairs_{\mathcal{S}}) \cup \{(\bar{E}, \bar{E}), (a, \bar{E}) \mid E \in \mathcal{X}, a \in Args_{\mathcal{S}} \setminus E\} \end{aligned}$$

where $\mathcal{X} = naive(F^{cf}_{\mathcal{S}}) \setminus \mathcal{S}$.

The idea of Construction 4.2 is to suitably extend the canonical framework $F^{cf}_{\mathcal{S}}$ from Construction 4.1 such that undesired stable and stage extensions are excluded. Coming back to our example with $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$, recall that $F^{cf}_{\mathcal{S}_3}$ (see Figure 3.1) had

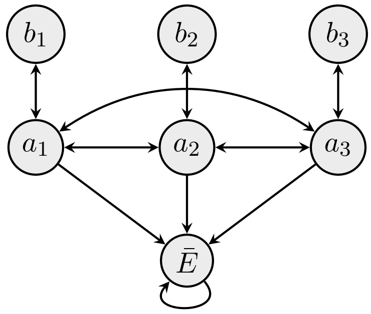

Figure 4.3: AF $F^{st}_{\mathcal{S}_3}$ with $stb(F^{st}_{\mathcal{S}_3}) = stage(F^{st}_{\mathcal{S}_3}) = \mathcal{S}_3$ where $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$

one such undesired stable and stage extension, $E = \{b_1, b_2, b_3\}$. To get rid of it we add a new argument which is attacked by all other arguments from $\mathcal{S}_3$ but not by E. Figure 4.3 shows this behaviour by depicting $F^{st}_{\mathcal{S}_3}$, which realizes $\mathcal{S}_3$ under the stable and stage semantics.

Proposition 4.4. *For each incomparable and tight extension-set $\mathcal{S}$, there exists an* AF *F such that $stb(F) = \mathcal{S}$. If $\mathcal{S} \neq \emptyset$, then $stb(F^{st}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. We have seen in Example 4.3 that the empty set is easily *stb*-realizable by the AF F_{prim} in Figure 4.2. In the following we deal with a non-empty $\mathcal{S}$. By Lemma 4.2 we already know that $\mathcal{S} \subseteq naive(F^{cf}_{\mathcal{S}})$. Let $\mathcal{X} = naive(F^{cf}_{\mathcal{S}}) \setminus \mathcal{S}$ and consider $F^{st}_{\mathcal{S}}$ from Construction 4.2. We show that $stb(F^{st}_{\mathcal{S}}) = \mathcal{S}$.

In order to show that $stb(F^{st}_{\mathcal{S}}) \subseteq \mathcal{S}$, let $E \in stb(F^{st}_{\mathcal{S}})$. As all new arguments (compared to $F^{cf}_{\mathcal{S}}$) $\bar{E}$ are self-attacking, also $naive(F^{st}_{\mathcal{S}}) = naive(F^{cf}_{\mathcal{S}})$ and therefore $E \in naive(F^{st}_{\mathcal{S}})$. Now as $naive(F^{st}_{\mathcal{S}}) = (\mathcal{X} \cup \mathcal{S})$ there are two options, either $E \in \mathcal{X}$ or $E \in \mathcal{S}$. Consider the first case, $E \in \mathcal{X}$. By construction of $F^{st}_{\mathcal{S}}$, $E \not\rightarrowtail \bar{E}$ and also $\bar{E} \notin E$, thus $E \notin stb(F^{st}_{\mathcal{S}})$, a contradiction. Hence it must hold that $E \in \mathcal{S}$, therefore $stb(F^{st}_{\mathcal{S}}) \subseteq \mathcal{S}$.

It remains to show that $stb(F^{st}_{\mathcal{S}}) \supseteq \mathcal{S}$. To this end consider some $E \in \mathcal{S}$. By Lemma 4.2, $E \in naive(F^{cf}_{\mathcal{S}})$, and, as $F^{cf}_{\mathcal{S}}$ is symmetric and irreflexive, we have $E \in stb(F^{cf}_{\mathcal{S}})$ by Proposition 2.15. Now consider $F^{st}_{\mathcal{S}}$. As we do not change attacks between the arguments $Args_{\mathcal{S}}$, $E \in naive(F^{st}_{\mathcal{S}})$ and E attacks all arguments in $Args_{\mathcal{S}} \setminus E$. Now consider an arbitrary argument $\bar{E}'$ for $E' \in \mathcal{X}$. $\bar{E}'$ is attacked by all arguments $a \in Args_{\mathcal{S}} \setminus E'$ and as E, E' are both naive sets (and thus incomparable) at least one of these arguments must be contained in E. Hence $E \in stb(F^{st}_{\mathcal{S}})$ follows. □

The following proposition shows that stable and stage semantics coincide in terms of realizability, with the exception of the empty set of extensions, which cannot be realized under the stage semantics. AFs realizing extension-sets under the stage semantics are also obtained from Construction 4.2.

Proposition 4.5. *For each non-empty, incomparable, and tight extension-set $\mathcal{S}$, it holds that $stage(F^{st}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. We know from Proposition 4.4 that $stb(F^{st}_{\mathcal{S}}) = \mathcal{S}$ for each non-empty, incomparable and tight extension-set $\mathcal{S}$. Since for each F with $stb(F) \neq \emptyset$, $stb(F) = stage(F)$ holds, it follows that $stage(F^{st}_{\mathcal{S}}) = \mathcal{S}$. □

Realizability of *adm*-based Semantics

When dealing with admissibility-based semantics, the concept of defense comes into play. This means that in order to realize an extension-set $\mathcal{S}$ under the admissibility-based semantics we consider in this thesis, i.e. admissible sets as well as preferred, semi-stable and complete semantics[2] by some AF F, the construction of this F has to ensure that every extension $S \in \mathcal{S}$ is defended by itself in F.

Towards such a suitable canonical AF for admissibility-based semantics we introduce the following technical concept.

Definition 4.2. Given an extension-set $\mathcal{S}$ and an argument $a \in Args_{\mathcal{S}}$, we define $\mathcal{S}^a = \{S \in \mathcal{S} \mid a \in S\}$ as the elements of $\mathcal{S}$ containing a. The *defense-formula* $Def^{\mathcal{S}}_a$ of argument a is $\top$ if $\{a\} \in \mathcal{S}$ and

$$\bigvee_{S \in \mathcal{S}^a} \bigwedge_{s \in S \setminus \{a\}} s$$

otherwise.

We call $Def^{\mathcal{S}}_a$ converted to a (logically equivalent) formula in conjunctive normal form *CNF-defense-formula* $CDef^{\mathcal{S}}_a$.

Intuitively, $Def^{\mathcal{S}}_a$ describes the conditions it takes for the argument a to be in an extension. The propositional atoms of $Def^{\mathcal{S}}_a$ (and $CDef^{\mathcal{S}}_a$) coincide with the arguments $Args_{\mathcal{S}}$. Each term represents a set of arguments which jointly allows a to "join" an extension. If $\mathcal{S}$ is a set of admissible extensions, each term represents the conjunction of arguments a is defended by.

Example 4.4. Consider the extension-set $\mathcal{S}_5 = \{\{a\}, \{b, c\}, \{a, c, d\}\}$. Then the defense-formulas are $Def^{\mathcal{S}_5}_a = \top$, $Def^{\mathcal{S}_5}_b = c$, $Def^{\mathcal{S}_5}_c = b \vee (a \wedge d)$ and $Def^{\mathcal{S}_5}_d = a \wedge c$. The corresponding CNF-defense-formulas are given as

- $CDef^{\mathcal{S}_5}_a = \{\}$,
- $CDef^{\mathcal{S}_5}_b = \{\{c\}\}$,
- $CDef^{\mathcal{S}_5}_c = \{\{a, b\}, \{b, d\}\}$, and
- $CDef^{\mathcal{S}_5}_d = \{\{a\}, \{c\}\}$.

[2] We also considered the grounded semantics, which is admissibility-based too. We do not take into account the grounded semantics in this chapter, since any extension-set, which contains exactly one element, is trivially realizable under the grounded semantics by the AF $F = (Args_{\mathcal{S}}, \emptyset)$. This also holds for the *ideal* semantics [34] and the *eager* semantics [22], which are also unique status semantics.

For extension-set $\mathcal{S}_6 = \{\{a\}, \{b\}, \{c\}, \{a, b, c\}\}$ we have the trivial case that all arguments out of $\mathit{Args}_{\mathcal{S}_6}$ form one-element-sets in $\mathcal{S}_6$. Therefore $\mathit{Def}_a^{\mathcal{S}_6} = \mathit{Def}_b^{\mathcal{S}_6} = \mathit{Def}_c^{\mathcal{S}_6} = \top$ and $\mathit{CDef}_a^{\mathcal{S}_6} = \mathit{CDef}_b^{\mathcal{S}_6} = \mathit{CDef}_c^{\mathcal{S}_6} = \{\}$

Finally considering $\mathcal{S}_7 = \{\{a, b\}, \{b, c\}, \{c, a\}\}$, the CNF-defense-formulas are

- $\mathit{CDef}_a^{\mathcal{S}_7} = \{\{b, c\}\}$,
- $\mathit{CDef}_b^{\mathcal{S}_7} = \{\{a, c\}\}$, and
- $\mathit{CDef}_c^{\mathcal{S}_7} = \{\{a, b\}\}$.

The following lemma shows that the (CNF-)defense-formula for any argument a really captures the intuition of describing which arguments it takes for a in order to join an element of the given extension-set.

Lemma 4.6. *Given an extension-set $\mathcal{S}$ and an argument $a \in \mathit{Args}_{\mathcal{S}}$, for each $S \subseteq \mathit{Args}_{\mathcal{S}}$ with $a \in S$ the following holds: $(S \setminus \{a\})$ is a model of $\mathit{Def}_a^{\mathcal{S}}$ (resp. $\mathit{CDef}_a^{\mathcal{S}}$) iff there exists an $S' \subseteq S$ with $a \in S'$ such that $S' \in \mathcal{S}$.*

Proof. The if-direction follows straight by definition of $\mathit{Def}_a^{\mathcal{S}}$. Let $S' \in \mathcal{S}$ with $a \in S'$, then the conjunction of the elements of $S' \setminus \{a\}$ forms a term of $\mathit{Def}_a^{\mathcal{S}}$. So $S' \setminus \{a\}$ is clearly a model of $\mathit{Def}_a^{\mathcal{S}}$.

To show the only-if-direction consider some $S \subseteq \mathit{Args}_{\mathcal{S}}$ with $a \in S$ where $S \setminus \{a\}$ is a model of $\mathit{Def}_a^{\mathcal{S}}$. If $\mathit{Def}_a^{\mathcal{S}} = \top$ then by Definition 4.2 it holds that $\{a\} \in \mathcal{S}$, and $S' = \{a\}$ fulfills the conditions. For $S \setminus \{a\}$ to be a model of $\mathit{Def}_a^{\mathcal{S}} \neq \top$, there must be some term $\tau \in \mathit{Def}_a^{\mathcal{S}}$, whose elements form a subset of $S \setminus \{a\}$. Consider such a term $\tau \in \mathit{Def}_a^{\mathcal{S}}$. Then by construction of $\mathit{Def}_a^{\mathcal{S}}$ there is some $S' \in \mathcal{S}$ with $a \in S'$, where $S' \setminus \{a\}$ coincides with the elements of τ. So $S' \subseteq S$.

Since $\mathit{Def}_a^{\mathcal{S}} \equiv \mathit{CDef}_a^{\mathcal{S}}$, these formulas can be used interchangeably in this proposition. □

Having at hand a formula for each argument, where its models coincide with the sets of arguments that defend this original argument, we can give the construction of our canonical defense-argumentation-framework.

Construction 4.3. Given an extension-set $\mathcal{S}$, we define the *canonical defense-argumentation-framework* by extending the canonical AF $F_{\mathcal{S}}^{cf} = (A_{\mathcal{S}}^{cf}, R_{\mathcal{S}}^{cf})$ to

$$F_{\mathcal{S}}^{def} = (A_{\mathcal{S}}^{def}, R_{\mathcal{S}}^{def})$$

where

$$A_{\mathcal{S}}^{def} = A_{\mathcal{S}}^{cf} \cup \bigcup_{a \in \mathit{Args}_{\mathcal{S}}} \{\alpha_{a,\gamma} \mid \gamma \in \mathit{CDef}_a^{\mathcal{S}}\}, \text{ and}$$
$$R_{\mathcal{S}}^{def} = R_{\mathcal{S}}^{cf} \cup \bigcup_{a \in \mathit{Args}_{\mathcal{S}}} \{(b, \alpha_{a,\gamma}), (\alpha_{a,\gamma}, \alpha_{a,\gamma}), (\alpha_{a,\gamma}, a) \mid \gamma \in \mathit{CDef}_a^{\mathcal{S}}, b \in \gamma\}.$$

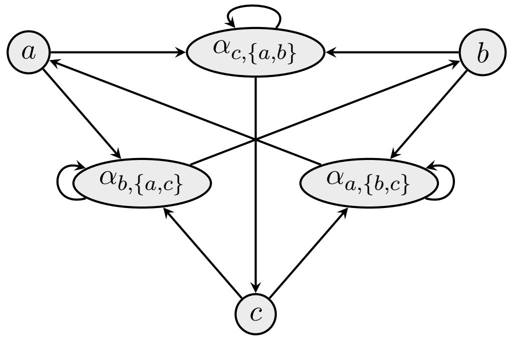

Figure 4.4: AF $F^{def}_{\mathcal{S}_7}$ for $\mathcal{S}_7 = \{\{a,b\},\{b,c\},\{c,a\}\}$

The canonical defense-argumentation-framework consists of all arguments given in the extension-set plus a certain amount of additional arguments, $\alpha_{a,\gamma}$. First of all each of these new arguments attacks itself in order not to be taken into account when it comes to figuring out the admissible sets of the framework. Further $\alpha_{a,\gamma}$ attacks argument a and is attacked by all arguments occurring as literals in the clause $\gamma \in \mathit{CDef}^{\mathcal{S}}_a$. So in this framework, for a to be defended from $\alpha_{a,\gamma}$, it takes at least one argument of these occurring as literals in the clause γ of $\mathit{CDef}^{\mathcal{S}}_a$. Therefore this constructed framework depicts the intended meaning of the defense-formulas, meaning that a certain argument needs other arguments to be defended.

First of all we will see that for any set of extensions $\mathcal{S}$, with the framework $F^{def}_{\mathcal{S}}$ from Construction 4.3 it is guaranteed that all $S \in \mathcal{S}$ form admissible sets. We motivate this by an example.

Example 4.5. Consider the extension-set $\mathcal{S}_6 = \{\{a\},\{b\},\{c\},\{a,b,c\}\}$ from Example 3.1. Note that $\mathcal{S}_6$ is not adm-closed, but comp-closed (see Table 3.1). The CNF-defense-formulas are given in Example 4.4. It is easy to verify that $R^{cf}_{\mathcal{S}}$ as well as $R^{def}_{\mathcal{S}}$ are empty and therefore $F^{def}_{\mathcal{S}_6} = (\{a,b,c\},\emptyset)$. So $\mathit{adm}(F^{def}_{\mathcal{S}_6}) = \mathit{dcl}(\mathcal{S}_6)$ and of course for all $S \in \mathcal{S}_6$ also $S \in \mathit{adm}(F^{def}_{\mathcal{S}_6})$. Still the other direction does not hold, because there are admissible sets of $F^{def}_{\mathcal{S}_6}$, namely $\emptyset$, $\{a,b\}$, $\{a,c\}$, and $\{b,c\}$, which are not in $\mathcal{S}_6$.

As another example consider the extension-set $\mathcal{S}_7 = \{\{a,b\},\{b,c\},\{c,a\}\}$, which is neither adm-closed nor comp-closed. Again the CNF-defense-formulas are given in Example 4.4. The constructed AF $F^{def}_{\mathcal{S}_7}$ is given by Figure 4.4. One can verify that $\mathit{adm}(F^{def}_{\mathcal{S}_7}) = (\mathcal{S}_7 \cup \{\{a,b,c\}\})$. So again it holds that $\mathcal{S}_7 \subseteq \mathit{adm}(F^{def}_{\mathcal{S}_7})$, but not the other direction since $\{a,b,c\} \notin \mathcal{S}_7$.

Actually, for both cases, $\mathcal{S}_6$ and $\mathcal{S}_7$, there is no AF, which realizes one of the extension-sets under the admissible semantics, as neither $\mathcal{S}_6$ nor $\mathcal{S}_7$ contains $\emptyset$. However, not even $(\mathcal{S}_6 \cup \{\emptyset\})$ nor $(\mathcal{S}_7 \cup \emptyset)$ is realizable under the admissible semantics, since we have seen in Proposition 3.16, that for each AF F it holds that $\mathit{adm}(F)$ is adm-closed.

We show that it holds for every extension-set, that each element of the extension-set is admissible in the AF obtained by Construction 4.3.

Proposition 4.7. *For each extension-set $\mathcal{S}$ it holds that $\mathcal{S} \subseteq \mathit{adm}(F^{def}_{\mathcal{S}})$.*

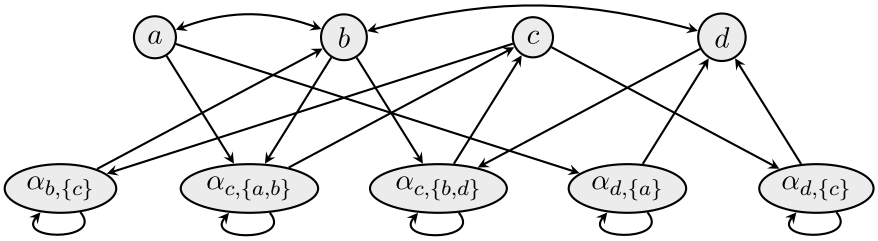

Figure 4.5: AF $F^{def}_{\mathcal{S}_5\cup\{\emptyset\}}$ with $adm(F^{def}_{\mathcal{S}_5\cup\{\emptyset\}} \cup \{\emptyset\}) = \mathcal{S}_5 \cup \{\emptyset\}$ where $\mathcal{S}_5 = \{\{a\}, \{b,c\}, \{a,c,d\}\}$

Proof. In order to show $\mathcal{S} \subseteq adm(F^{def}_{\mathcal{S}})$, consider an arbitrary $S \in \mathcal{S}$. If $S = \emptyset$, the assertion trivially holds. If S consists of exactly one argument, i.e. $S = \{a\}$, then $CDef^{\mathcal{S}}_a$ is the empty set of clauses. By definition of $R^{def}_{\mathcal{S}}$, a is then defended in $F^{def}_{\mathcal{S}}$ and thus $S \in adm(F^{def}_{\mathcal{S}})$. Thus let $S \in \mathcal{S}$ contain at least two arguments. By construction, S is conflict-free in $F^{def}_{\mathcal{S}}$, since conflicts between two original arguments s_1, s_2 are only constructed if $(s_1, s_2) \notin Pairs_{\mathcal{S}}$, which is not the case since $S \in \mathcal{S}$.

It remains to show that each $s \in S$ is defended by S in $F^{def}_{\mathcal{S}}$. Let $s \in S$. If $\{s\} \in \mathcal{S}$, we know from the one-element case that $\{s\} \in adm(F^{def}_{\mathcal{S}})$, so s is defended. On the other hand, assume $\{s\} \notin \mathcal{S}$. By Lemma 4.6 we know that $S \setminus \{s\}$ is a model of $Def^{\mathcal{S}}_s$ as well as of $CDef^{\mathcal{S}}_s$. The latter means that each clause $\gamma \in CDef^{\mathcal{S}}_s$ contains at least one variable $t_\gamma \in S \setminus \{s\}$. Thus, by construction of $R^{def}_{\mathcal{S}}$, it follows that $S \setminus \{s\} \mapsto \alpha_{s,\gamma}$ for each $\gamma \in CDef^{\mathcal{S}}_s$, i.e. s is defended by $S \setminus \{s\}$ in $F^{def}_{\mathcal{S}}$. Hence $S \in adm(F^{def}_{\mathcal{S}})$ and therefore $\mathcal{S} \subseteq adm(F^{def}_{\mathcal{S}})$. □

Knowing that $\mathcal{S} \subseteq adm(F^{def}_{\mathcal{S}})$ holds for any extension-set $\mathcal{S}$, we will show now that the other direction, i.e. $adm(F^{def}_{\mathcal{S}}) \subseteq \mathcal{S}$, holds for all adm-closed extension-sets, meaning that such sets are realizable under the admissible semantics.

Again we begin with an example.

Example 4.6. Again consider the extension-set $\mathcal{S}_5$, but extended by the empty extension. As $\emptyset$ is an admissible set in any AF, we need it in the extension-set in order to find an AF realizing the extension-set. So we have $\mathcal{S}_5 \cup \{\emptyset\} = \{\emptyset, \{a\}, \{b,c\}, \{a,c,d\}\}$. Note that adding $\emptyset$ does not make a difference with the (CNF-)defense-formulas. We have given the CNF-defense-formulas in Example 4.4. $F^{def}_{\mathcal{S}_5\cup\{\emptyset\}}$ thus is given by the AF in Figure 4.5. Considering, for instance, argument c, having $CDef^{\mathcal{S}_5\cup\{\emptyset\}}_c = \{\{a,b\},\{b,d\}\}$, one can see that in $F^{def}_{\mathcal{S}_5\cup\{\emptyset\}}$ it takes a or b in order to defend c from $\alpha_{c,\{a,b\}}$, and b or d in order to defend c from $\alpha_{c,\{b,d\}}$. Long story short, $adm(F^{def}_{\mathcal{S}_5\cup\{\emptyset\}}) = (\mathcal{S}_5 \cup \{\emptyset\})$.

Proposition 4.8. *For each adm-closed extension-set $\mathcal{S}$ containing $\emptyset$, it holds that $adm(F^{def}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. We have already seen in Proposition 4.7, that $adm(F_{\mathcal{S}}^{def}) \supseteq \mathcal{S}$ holds for every extension-set $\mathcal{S}$.

So it remains to show that $adm(F_{\mathcal{S}}^{def}) \subseteq \mathcal{S}$ holds for each adm-closed extension-set $\mathcal{S}$ containing $\emptyset$. Consider some $S \in adm(F_{\mathcal{S}}^{def})$. First of all, S cannot contain any of the self-attacking arguments $\alpha_{a,\gamma}$. For $S = \emptyset$, $S \in \mathcal{S}$ holds by definition. If S consists of exactly one argument, i.e. $S = \{a\}$, it must hold that $\forall b \in A \; s.t. \; b \mapsto a : \; a \mapsto b$. For that, by construction of $F_{\mathcal{S}}^{def}$, it must hold that $CDef_a^{\mathcal{S}} = \{\}$, therefore $S \in \mathcal{S}$. Now assume S contains at least two arguments. S being conflict-free, by definition of $R_{\mathcal{S}}^{def}$, guarantees that $\forall a, b \in S : \; (a, b) \in Pairs_{\mathcal{S}}$. Let $s \in S$ with $\{s\} \notin adm(F_{\mathcal{S}}^{def})$. Then we have $\alpha_{s,\gamma} \mapsto s$ for each $\gamma \in CDef_s^{\mathcal{S}}$. Since s is defended by S, for each $\gamma \in CDef_s^{\mathcal{S}}$, it holds that $\exists t_\gamma \in (S \setminus \{s\}) : \; t_\gamma \mapsto \alpha_{s,\gamma}$. By definition, thus t_γ occurs in the clause γ. It follows that $T = \{t_\gamma \mid \gamma \in CDef_s^{\mathcal{S}}\}$ is a model of $CDef_s^{\mathcal{S}}$ and thus also of the defense-formula $Def_s^{\mathcal{S}}$. Then by Lemma 4.6 there is some $S' \subseteq T \cup \{s\}$ with $s \in S'$ such that $S' \in \mathcal{S}$. Recall also that in case $\{s\} \in adm(F_{\mathcal{S}}^{def})$, we know from above that $\{s\} \in \mathcal{S}$. Connecting these conclusions with the facts that $\mathcal{S}$ is adm-closed and $\forall a, b \in S : \; (a, b) \in Pairs_{\mathcal{S}}$, we get $S \in \mathcal{S}$. □

We can use the same construction to realize certain, namely non-empty and pref-closed, extension-sets under the preferred semantics.

Example 4.7. In Example 4.6 we have seen that $adm(F_{\mathcal{S}_5}^{def} \cup \{\emptyset\}) = (\mathcal{S}_5 \cup \{\emptyset\})$. As $\mathcal{S}_5$ is not pref-closed consider the pref-closed extension-set $\mathcal{S}_5' = \mathcal{S}_5 \setminus \{\{a\}\} = \{\{b, c\}, \{a, c, d\}\}$. We get the following CNF-defense-formulas:

- $CDef_a^{\mathcal{S}_5'} = \{\{c\}, \{d\}\}$,
- $CDef_b^{\mathcal{S}_5'} = \{\{c\}\}$,
- $CDef_c^{\mathcal{S}_5'} = \{\{a, b\}, \{b, d\}\}$, and
- $CDef_d^{\mathcal{S}_5'} = \{\{a\}, \{c\}\}$.

Thus $F_{\mathcal{S}_5'}^{def}$ is given by the AF in Figure 4.6. We observe that $pref(F_{\mathcal{S}_5'}^{def}) = (adm(F_{\mathcal{S}_5'}^{def}) \setminus \{\emptyset\}) = \mathcal{S}_5'$.

Now we are ready to show that the AFs one obtains from Construction 4.3 are also suitable to realize non-empty, pref-closed extension-sets under the preferred semantics.

Proposition 4.9. *For each non-empty and pref-closed extension-set $\mathcal{S}$, it holds that $pref(F_{\mathcal{S}}^{def}) = \mathcal{S}$.*

Proof. Let $\mathcal{S}$ be a non-empty and pref-closed extension-set and $\mathcal{S}' = (\mathcal{S} \cup \{\emptyset\})$. Clearly $\mathcal{S}'$ is non-empty. By Construction 4.3 it holds that $F_{\mathcal{S}'}^{def} = F_{\mathcal{S}}^{def}$. From Lemma 3.9 we obtain that $\mathcal{S}'$ is adm-closed. Then by Proposition 4.8 $adm(F_{\mathcal{S}}^{def}) = \mathcal{S}'$. As preferred extensions are $\subseteq$-maximal admissible sets and since $\mathcal{S}$ is incomparable by Lemma 3.8, $pref(F_{\mathcal{S}}^{def}) = \mathcal{S}$ follows. □

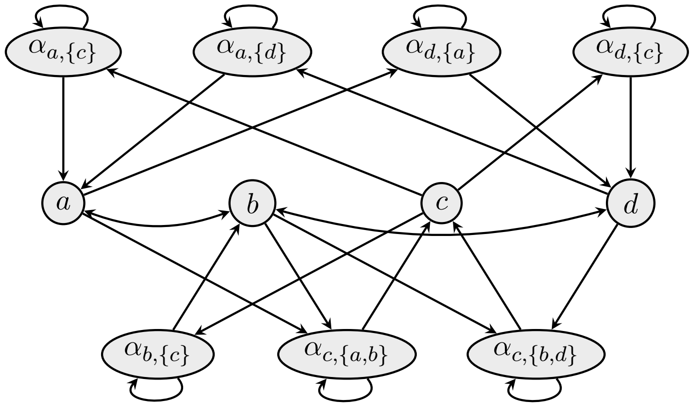

Figure 4.6: AF $F^{def}_{\mathcal{S}'_5}$ where $\mathcal{S}'_5 = \mathcal{S}_5 \setminus \{\{a\}\}$ with $pref(F^{def}_{\mathcal{S}'_5}) = (adm(F^{def}_{\mathcal{S}'_5}) \setminus \{\emptyset\}) = \mathcal{S}'_5$

Another way to characterize the *pref*-realizable extension-sets is the following.

Corollary 4.10. *For each non-empty, incomparable and adm-closed extension-set $\mathcal{S}$, it holds that pref*$(F^{def}_{\mathcal{S}}) = \mathcal{S}$.

Proof. This follows immediately from Proposition 4.9 and Lemma 3.8. □

Coming to semi-stable semantics we have seen by Proposition 3.19 and Proposition 3.17 that extension-sets under the semi-stable and the preferred semantics share the same necessary properties. We will show that being non-empty and pref-closed are also sufficient conditions for an extension-set to be realizable under the semi-stable semantics. Therefor we will need a minor adjustment to Construction 4.3 in order to deal with the range of extensions in the constructed AF. The following construction is inspired by the first translation in [39], just the symmetry of the additional attacks is omitted.

Construction 4.4. Given an extension-set $\mathcal{S}$, we define the *enhanced canonical defense-argumentation-framework* by extending the canonical defense-AF $F^{def}_{\mathcal{S}} = (A^{def}_{\mathcal{S}}, R^{def}_{\mathcal{S}})$ to

$$F^{sem}_{\mathcal{S}} = (A^{sem}_{\mathcal{S}}, R^{sem}_{\mathcal{S}}),$$

where

$$A^{sem}_{\mathcal{S}} = A^{def}_{\mathcal{S}} \cup \bigcup_{a \in Args_{\mathcal{S}}} \{\delta_a\}, \text{ and}$$

$$R^{sem}_{\mathcal{S}} = R^{def}_{\mathcal{S}} \cup \bigcup_{a \in Args_{\mathcal{S}}} \{(a, \delta_a), (\delta_a, \delta_a)\}.$$

The introduction of such a self-attacking argument δ_a for each $a \in Args_{\mathcal{S}}$ which is attacked by a has the effect that for two different extensions $A, B \in \mathcal{S}$, where $A \not\subseteq B$ holds, also $A^+ \not\subseteq B^+$ holds.

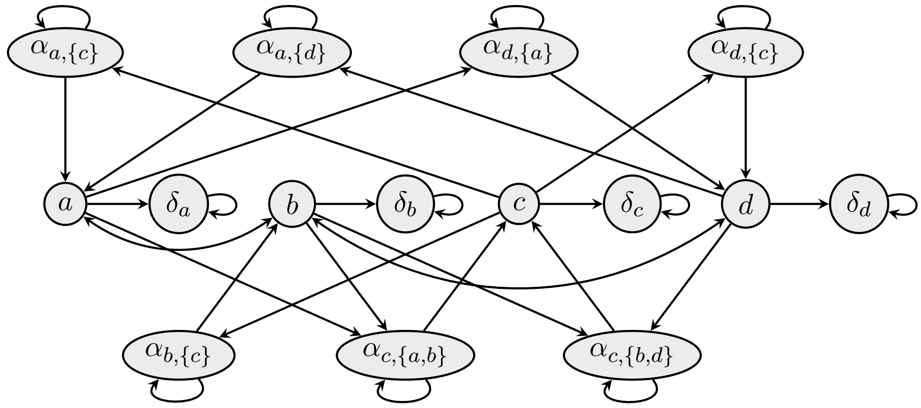

Figure 4.7: AF $F^{sem}_{\mathcal{S}'_5}$ where $\mathcal{S}'_5 = \mathcal{S}_5 \setminus \{\{a\}\}$ with $sem(F^{sem}_{\mathcal{S}'_5}) = \mathcal{S}'_5$

We look at an example before proposing the realizability-result.

Example 4.8. In Example 4.7 we have seen that $pref(F^{def}_{\mathcal{S}'_5}) = \mathcal{S}'_5$, where $\mathcal{S}'_5 = (\mathcal{S}_5 \setminus \{\{a\}\}) = \{\{b,c\}, \{a,c,d\}\}$. Observe that in $F^{def}_{\mathcal{S}'_5} = (A^{def}_{\mathcal{S}'_5}, R^{def}_{\mathcal{S}'_5})$, $\{a,c,d\}^+ = A^{def}_{\mathcal{S}'_5}$, but $\{b,c\}^+ = A^{def}_{\mathcal{S}'_5} \setminus \{\alpha_{a,\{d\}}, \alpha_{d,\{a\}}\}$. Therefore $\{b,c\} \notin sem(F^{def}_{\mathcal{S}'_5})$.

Now consider $F^{sem}_{\mathcal{S}'_5}$ in Figure 4.7. As each argument $x \in Args_{\mathcal{S}'_5}$ exclusively attacks some δ_x, the range of the two extensions is incomparable, i.e. $\delta_b \notin \{a,c,d\}^+$ and, for instance, $\delta_a \notin \{b,c\}^+$. Therefore $sem(F^{sem}_{\mathcal{S}'_5}) = \mathcal{S}'_5$.

Proposition 4.11. *For each non-empty, pref-closed extension-set $\mathcal{S}$, it holds that $sem(F^{sem}_{\mathcal{S}}) = \mathcal{S}$.*

Proof. First note that since each argument newly introduced in Construction 4.4, i.e. δ_a for $a \in Args_{\mathcal{S}}$, attacks itself and does not attack any $a \in A^{sem}_{\mathcal{S}}$, it follows that $adm(F^{def}_{\mathcal{S}}) = adm(F^{sem}_{\mathcal{S}})$ and therefore also $pref(F^{def}_{\mathcal{S}}) = pref(F^{sem}_{\mathcal{S}})$ for each extension-set $\mathcal{S}$.

As, for a non-empty and pref-closed extension-set $\mathcal{S}$, we know $pref(F^{def}_{\mathcal{S}}) = \mathcal{S}$ and also $pref(F^{sem}_{\mathcal{S}}) = \mathcal{S}$, $sem(F^{sem}_{\mathcal{S}}) \subseteq \mathcal{S}$ follows from the fact that $sem(F) \subseteq pref(F)$ for each AF F (cf. Proposition 2.13).

It remains to show that $\mathcal{S} \subseteq sem(F^{sem}_{\mathcal{S}})$. Consider some $A, B \in \mathcal{S}$ and to the contrary assume that $A^+ \subset B^+$, meaning that $A \notin sem(F^{sem}_{\mathcal{S}})$. Since $A, B \in pref(F^{sem}_{\mathcal{S}})$ it holds that $A \not\subseteq B$. Therefore there exists some $a \in A \setminus B$. But then, by definition of $F^{sem}_{\mathcal{S}}$ in Construction 4.4, $\delta_a \in A^+$, but $\delta_a \notin B^+$, a contradiction to $A^+ \subset B^+$. Therefore for each $A \in \mathcal{S}$ it holds that $A \in sem(F^{sem}_{\mathcal{S}})$, i.e. $\mathcal{S} \subseteq sem(F^{sem}_{\mathcal{S}})$. □

Now we turn to the complete semantics. When constructing an AF in order to realize some extension-set under the complete semantics we not only have to take care of conflict-freeness and defense of each element of the extension-set, but also have to ensure that the extensions we want to realize are fixed-points of the characteristic function of the constructed AF. To this end we first introduce another technical concept.

Definition 4.3. Given a comp-closed extension-set $\mathcal{S}$ and an argument $a \in \mathit{Args}_{\mathcal{S}}$, let $\mathcal{T}^a = \{A \cup B \mid A, B \in \mathcal{S}, \forall s, t \in A \cup B : (s,t) \in \mathit{Pairs}_{\mathcal{S}}, a \notin (A \cup B), a \in C_{\mathcal{S}}(A \cup B)\}$[3]. We define the *completion-formula* $\mathit{Com}_a^{\mathcal{S}}$ of argument a as $\top$ if $\mathcal{T}^a = \emptyset$ and

$$\bigvee_{S \in \mathcal{T}^a} \bigwedge_{s \in S} s$$

otherwise.

$\mathit{Com}_a^{\mathcal{S}}$ converted to a (logically equivalent) formula in conjunctive normal form we call *CNF-completion-formula* $\mathit{CCom}_a^{\mathcal{S}}$.

Intuitively, the completion-formula of an argument $a \in \mathit{Args}_{\mathcal{S}}$ describes the sets of arguments which have to be "completed" by a. The idea behind the completion-formula is when building an AF F on the basis of some comp-closed extension-set $\mathcal{S}$, where some sets $S \notin \mathcal{S}$ are admissible one may need a concept to let these admissible sets in F defend other arguments in order not to be complete extensions. In particular when dealing with a comp-closed extension-set, one may have $A, B \in \mathcal{S}$, but $(A \cup B) \notin \mathcal{S}$ (some unique $C \supseteq (A \cup B)$ with $C \in \mathcal{S}$ though). In any AF F where A, B, and C are admissible, also $A \cup B$ is, so you need to force $A \cup B$ to defend some $a \in C \setminus (A \cup B)$ in order not to be a complete extension.

Example 4.9. Consider the extension-set $\mathcal{S}_6 = \{\{a\}, \{b\}, \{c\}, \{a, b, c\}\}$. Then the completion-formulas are $\mathit{Com}_a^{\mathcal{S}_6} = b \wedge c$, $\mathit{Com}_b^{\mathcal{S}_6} = a \wedge c$ and $\mathit{Com}_c^{\mathcal{S}_6} = a \wedge b$. The corresponding CNF-completion-formulas are given as

- $\mathit{CCom}_a^{\mathcal{S}_6} = \{\{b\}, \{c\}\}$,
- $\mathit{CCom}_b^{\mathcal{S}_6} = \{\{a\}, \{c\}\}$, and
- $\mathit{CCom}_c^{\mathcal{S}_6} = \{\{a\}, \{b\}\}$.

We are now ready to define the canonical completion-argumentation-framework, which we will use in order to realize extension-sets fulfilling sufficient conditions under the complete semantics.

Construction 4.5. Given an extension-set $\mathcal{S}$, let $\mathit{Gr} = \bigcap_{S \in \mathcal{S}} S$. We define the *canonical completion-argumentation-framework* by extending the canonical AF $F_{\mathcal{S}}^{cf} = (A_{\mathcal{S}}^{cf}, R_{\mathcal{S}}^{cf})$ to

$$F_{\mathcal{S}}^{comp} = (A_{\mathcal{S}}^{cf} \cup A_{\mathcal{S}}^{\prime def} \cup A_{\mathcal{S}}^{comp} \cup A_{\mathcal{S}}^{g}, R_{\mathcal{S}}^{cf} \cup R_{\mathcal{S}}^{\prime def} \cup R_{\mathcal{S}}^{comp} \cup R_{\mathcal{S}}^{g})$$

where

[3] Note that as $\mathcal{S}$ is comp-closed, $\mathcal{C}_{\mathcal{S}}(A \cup B)$ contains exactly one element, and therefore $C_{\mathcal{S}}(A \cup B)$ is well-defined

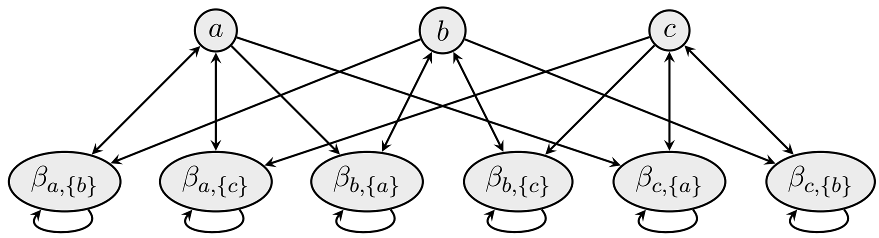

Figure 4.8: AF $F^{comp}_{\mathcal{S}'_6}$ with $comp(F^{comp}_{\mathcal{S}'_6}) = \mathcal{S}'_6$ where $\mathcal{S}'_6 = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b, c\}\}$

$$
\begin{aligned}
A'^{def}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr)} \{\alpha_{a,\gamma} \mid \gamma \in CDef^{\mathcal{S}}_a\},\\
A^{comp}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr)} \{\beta_{a,\gamma} \mid \gamma \in CCom^{\mathcal{S}}_a\},\\
A^{g}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr), CCom^{\mathcal{S}}_a = \top} \{\delta_a\},\\
R'^{def}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr)} \{(b, \alpha_{a,\gamma}), (\alpha_{a,\gamma}, \alpha_{a,\gamma}), (\alpha_{a,\gamma}, a) \mid \gamma \in CDef^{\mathcal{S}}_a, b \in \gamma\},\\
R^{comp}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr)} \{(b, \beta_{a,\gamma}), (\beta_{a,\gamma}, \beta_{a,\gamma}), (\beta_{a,\gamma}, a), (a, \beta_{a,\gamma}) \mid \gamma \in CCom^{\mathcal{S}}_a, b \in \gamma\}, \text{ and}\\
R^{g}_{\mathcal{S}} &= \bigcup_{a \in (Args_{\mathcal{S}} \setminus Gr), CCom^{\mathcal{S}}_a = \top} \{(a, \delta_a), (\delta_a, \delta_a), (\delta_a, a)\}.
\end{aligned}
$$

Note that in $F^{comp}_{\mathcal{S}}$ from Construction 4.5, each argument $a \in Gr$ has no ingoing attacks.

By looking at Construction 4.5 in a procedural way, one could split the procedure into three steps.

1. In a first step, with $R^{cf}_{\mathcal{S}}$ and $R'^{def}_{\mathcal{S}}$, the AF is build in a way that all $S \in \mathcal{S}$ are admissible sets in $F^{comp}_{\mathcal{S}}$. This works nearly the same way as in Construction 4.3. Still there can be admissible sets in $F^{comp}_{\mathcal{S}}$ which are not in $\mathcal{S}$.

2. In the second step, by adding $R^{comp}_{\mathcal{S}}$, admissible sets $A \cup B$ where $A, B \in \mathcal{S}$, but $(A \cup B) \notin \mathcal{S}$, are forced to defend certain arguments $a \in C_{\mathcal{S}}(A \cup B)$ to ensure that $(A \cup B) \notin comp(F^{comp}_{\mathcal{S}})$.

3. In the third and last step, adding $R^{g}_{\mathcal{S}}$ makes sure that the remaining admissible sets are also complete extensions, since it is impossible for them to defend any argument a, where $\delta_a \mapsto a$ and only $a \mapsto \delta_a$.

In none of the steps, any attack on an argument $a \in Gr$ is added in order to let Gr surely be the least complete extension.

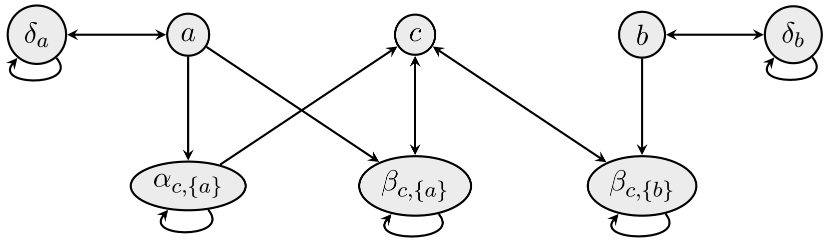

Figure 4.9: AF $F_{\mathcal{S}}^{comp}$ with $comp(F_{\mathcal{S}}^{comp}) = \mathcal{S}$ where $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{a, c\}, \{a, b, c\}\}$

Example 4.10. Consider the extension-set $\mathcal{S}_6' = \mathcal{S}_6 \cup \{\emptyset\} = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b, c\}\}$. We have given the CNF-completion-formulas in Example 4.9 (note that $CCom_s^{\mathcal{S}_6} = CCom_s^{\mathcal{S}_6'}$ for any $s \in Args_{\mathcal{S}_6}$). Further $CDef_s^{\mathcal{S}_6'} = \{\}$ for any $s \in Args_{\mathcal{S}_6}$ (see Example 4.4). $F_{\mathcal{S}_6'}^{comp}$ thus is given by the AF in Figure 4.8. Consider, for instance, $\{a\} \in \mathcal{S}_6'$ and $\{b\} \in \mathcal{S}_6'$, where $(\{a\} \cup \{b\}) \notin \mathcal{S}_6'$. It is easy to see that $\{a\} \cup \{b\}$ is admissible in $F_{\mathcal{S}_6'}^{comp}$, but as $\{a\} \cup \{b\}$ defends arguments c from its attackers $\beta_{c,\{a\}}$ and $\beta_{c,\{b\}}$, it follows that $\{a, b\}$ is not a complete extension of $F_{\mathcal{S}_6'}^{comp}$, but $\{a, b, c\}$ is. The same holds equivalently for $\{b, c\}$ and $\{a, c\}$.

In $F_{\mathcal{S}_6'}^{comp}$ from Example 4.10, it holds that $R_{\mathcal{S}_6'}^{def} = R_{\mathcal{S}_6'}^{g} = \emptyset$. To get a better understanding of the construction, we give another example.

Example 4.11. Consider the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{a, c\}, \{a, b, c\}\}$. The CNF-defense-formulas are $CDef_a^{\mathcal{S}} = CDef_b^{\mathcal{S}} = \{\}$ and $CDef_c^{\mathcal{S}} = \{\{a\}\}$. The CNF-completion-formulas are given by $CCom_a^{\mathcal{S}} = CCom_b^{\mathcal{S}} = \{\}$ and $CCom_c^{\mathcal{S}} = \{\{a\}, \{b\}\}$. Hence the AF $F_{\mathcal{S}}^{comp}$ is the one depicted in Figure 4.9.

Proposition 4.12. *For each non-empty and comp-closed extension-set $\mathcal{S}$ which contains $Gr = \bigcap_{S \in \mathcal{S}} S$, it holds that $comp(F_{\mathcal{S}}^{comp}) = \mathcal{S}$.*

Proof. We begin by showing $\boldsymbol{\mathcal{S} \subseteq comp(F_{\mathcal{S}}^{comp})}$: Consider some $S \in \mathcal{S}$. By Proposition 4.7 we know that $\mathcal{S} \subseteq adm(F_{\mathcal{S}}^{def})$, i.e. S is admissible in $F_{\mathcal{S}}^{def}$. Note that all attackers $\beta_{a,\gamma} \in A_{\mathcal{S}}^{comp}$ and $\delta_a \in A_{\mathcal{S}}^{g}$ (these are just the additional arguments in $F_{\mathcal{S}}^{comp}$ compared to $F_{\mathcal{S}}^{def}$) are re-attacked by the argument they are attacking, which means that any argument is defended by itself from any of these attackers. Also these arguments are self-attacking. Therefore S is admissible in $F_{\mathcal{S}}^{comp}$.

It remains to show that S is a complete extension of $F_{\mathcal{S}}^{comp}$, i.e. for all a defended by S, $a \in S$. To the contrary, assume some $a \in Args_{\mathcal{S}} \setminus S$ is defended by S in $F_{\mathcal{S}}^{comp}$. Note that $S \cup \{a\}$ is then admissible in $F_{\mathcal{S}}^{comp}$ by Dung's fundamental lemma (Lemma 2.3). Further note that $a \notin Gr$ since $Gr \subseteq S$. We distinguish between the cases that either there is some $\beta_{a,\gamma} \in A_{\mathcal{S}}^{comp}$ attacking a or not.

1. Consider the case that $\nexists \beta_{a,\gamma} \in A_{\mathcal{S}}^{comp} : (\beta_{a,\gamma}, a) \in R_{\mathcal{S}}^{comp}$. Hence $CCom_{\mathcal{S}}^{a} = Com_{\mathcal{S}}^{a} = \top$. But then $\exists \delta_a \in A_{\mathcal{S}}^{g} : (\delta_a, a), (a, \delta_a) \in R_{\mathcal{S}}^{g}$, and δ_a having no additional attackers, a contradiction to the assumption that a is defended by S in $F_{\mathcal{S}}^{comp}$.

2. Now consider the case that $\exists \beta_{a,\gamma} \in A^{comp}_{\mathcal{S}} : (\beta_{a,\gamma}, a) \in R^{comp}_{\mathcal{S}}$. In order to defend a, S has to attack $\beta_{a,\gamma}$ for each $\gamma \in CCom^{\mathcal{S}}_a$. Then by Construction 4.5 $S \models CCom^{\mathcal{S}}_a$, and also $S \models Com^{\mathcal{S}}_a$. The latter means by Definition 4.3 that $\exists A, B \in \mathcal{S} : a \notin (A \cup B) \wedge a \in C_{\mathcal{S}}(A \cup B) \wedge S \supseteq (A \cup B)$. Consider such $A, B \in \mathcal{S}$. Since $a \in C_{\mathcal{S}}(A \cup B)$ but $a \notin S$, it follows that $S \not\supseteq C_{\mathcal{S}}(A \cup B)$. If $S \subset C_{\mathcal{S}}(A \cup B)$, we get a contradiction to $\mathcal{S}$ being comp-closed. The same holds in the last case, where S and $C_{\mathcal{S}}(A \cup B)$ are incomparable, since $C_{\mathcal{S}}(A \cup B)$ is then not a unique superset of $A \cup B$.

It remains to show that $\boldsymbol{comp}(\mathbf{F}^{\boldsymbol{comp}}_{\boldsymbol{\mathcal{S}}}) \subseteq \boldsymbol{\mathcal{S}}$: First of all, any $S \in comp(F^{comp}_{\mathcal{S}})$ cannot contain any of the self-attacking arguments $\alpha_{a,\gamma} \in A'^{def}_{\mathcal{S}}$, $\beta_{a,\gamma} \in A^{comp}_{\mathcal{S}}$ and $\delta_a \in A^g_{\mathcal{S}}$.

- We begin by considering the grounded extension of $F^{comp}_{\mathcal{S}}$, $grd(F^{comp}_{\mathcal{S}})$. We argue that $grd(F^{comp}_{\mathcal{S}}) = Gr$, where $Gr = \bigcap_{S \in \mathcal{S}} S$. Note that Gr is contained in $\mathcal{S}$ by assumption. Since $Gr \subseteq S$ holds for all $S \in \mathcal{S}$, there are no $a \in Args_{\mathcal{S}}$ and $b \in Gr$ such that $(a, b) \in R^{cf}_{\mathcal{S}}$ or $(b, a) \in R^{cf}_{\mathcal{S}}$. Also by Construction 4.5 there are no attacks on any $b \in Gr$ in any of $R'^{def}_{\mathcal{S}}$, $R^{comp}_{\mathcal{S}}$ and $R^g_{\mathcal{S}}$. Therefore $Gr \subseteq grd(F^{comp}_{\mathcal{S}})$. Particularly, Gr is the set of arguments in $F^{comp}_{\mathcal{S}}$ with no ingoing attacks. We continue by showing that $Gr \supseteq grd(F^{comp}_{\mathcal{S}})$. Note that for each argument $s \in Args_{\mathcal{S}}$ with $s \notin Gr$, there exists an $a \in (A^g_{\mathcal{S}} \cup A^{comp}_{\mathcal{S}})$ such that (a, s) either in $R^g_{\mathcal{S}}$ or in $R^{comp}_{\mathcal{S}}$. Consider such $s \in Args_{\mathcal{S}} \setminus Gr$ and $a \in (A^g_{\mathcal{S}} \cup A^{comp}_{\mathcal{S}})$ and, towards a contradiction, assume s is defended by Gr, meaning $s \in grd(F^{comp}_{\mathcal{S}})$. If $(a, s) \in R^g_{\mathcal{S}}$, we get a contradiction to Gr defending s, since a is only attacked by s and by itself. So $(a, s) \in R^{comp}_{\mathcal{S}}$. Then Gr must defend s from each attacking a, i.e. $Gr \models CCom^{\mathcal{S}}_s$, and also $Gr \models Com^{\mathcal{S}}_s$. Therefore by Construction 4.5 there must be some term $\tau \in Com^{\mathcal{S}}_s$ with $Gr \supseteq \tau$. This means by Definition 4.3 there are some $A, B \in \mathcal{S}$ with $(A \cup B) \notin \mathcal{S}$ and $Gr \supseteq (A \cup B)$. But this is a contradiction to $Gr \in \mathcal{S}$ as for all $S \in \mathcal{S}$, $S \supseteq Gr$. Therefore $Gr \subseteq grd(F^{comp}_{\mathcal{S}})$ and also $grd(F^{comp}_{\mathcal{S}}) = Gr$.

- Finally we show that for each $S \in (comp(F^{comp}_{\mathcal{S}}) \setminus \{Gr\})$, it holds that $S \in \mathcal{S}$. We begin by arguing that $\mathcal{C}(A \cup B) \subseteq S$ for each $S \in (comp(F^{comp}_{\mathcal{S}}) \setminus \{Gr\})$ and $A, B \subseteq S$, where $A, B \in \mathcal{S}$. To this end let $S \in (comp(F^{comp}_{\mathcal{S}}) \setminus \{Gr\})$ and $A, B \subseteq S$ with $A, B \in \mathcal{S}$. As S is a complete extension, for all $s \in Args_{\mathcal{S}} \setminus S$[4] it holds that s is not defended by S. Consider such an $s \in Args_{\mathcal{S}} \setminus S$. By Construction 4.5 each argument is attacked either by some $a \in A^g_{\mathcal{S}}$ or $a \in A^{comp}_{\mathcal{S}}$.

  1. Assume there is some $a \in A^g_{\mathcal{S}}$ such that $(a, s) \in R^g_{\mathcal{S}}$. Then $CCom^{\mathcal{S}}_s = \{\}$, which means by Definition 4.3 that $\nexists A', B' \in \mathcal{S} : s \notin (A' \cup B'), s \in C_{\mathcal{S}}(A' \cup B')$. Hence also $s \notin C_{\mathcal{S}}(A \cup B)$.
  2. Assume there is some $a \in A^{comp}_{\mathcal{S}}$ such that $(a, s) \in R^{comp}_{\mathcal{S}}$ and $S \mapsto a$ does not hold. Then by Construction 4.5 $S \not\models Com^{\mathcal{S}}_s$ and also $S \not\models CCom^{\mathcal{S}}_s$. Hence $S \not\supseteq \tau$ for any term $\tau \in Com^{\mathcal{S}}_s$. This means by Definition 4.3 that $\nexists A', B' \in \mathcal{S} : s \notin A' \cup B' \wedge s \in C_{\mathcal{S}}(A' \cup B') \wedge S \supseteq (A' \cup B')$. Hence, again $s \notin C_{\mathcal{S}}(A \cup B)$.

[4] Again it suffices to consider $Args_{\mathcal{S}}$ since all other arguments in $F^{comp}_{\mathcal{S}}$ are self-attacking.

Therefore $C_{\mathcal{S}}(A \cup B) \subseteq S$ for each $S \in (comp(F_{\mathcal{S}}^{comp}) \setminus \{Gr\})$ and $A, B \subseteq S$, where $A, B \in \mathcal{S}$.

Since S is a complete extension also $S \in adm(F_{\mathcal{S}}^{comp})$. So each $s \in S$ is defended by S. Consider some $s \in S$. By Construction 4.5, s defends itself from all possible attackers in $A_{\mathcal{S}}^{cf}$, $A_{\mathcal{S}}^{comp}$ and $A_{\mathcal{S}}^{g}$. So we take a look at the attackers in $A_{\mathcal{S}}^{\prime def}$. We have $\alpha_{s,\gamma} \mapsto s$ for each $\gamma \in CDef_s^{\mathcal{S}}$. Since s is defended by S, for each $\gamma \in CDef_s^{\mathcal{S}}$, it holds that $\exists t_\gamma \in (S \setminus \{s\}) : t_\gamma \mapsto \alpha_{s,\gamma}$. By definition, thus t_γ occurs in the clause γ. It follows that $T = \{t_\gamma \mid \gamma \in CDef_s^{\mathcal{S}}\}$ is a model of $CDef_s^{\mathcal{S}}$ and thus also of the defense-formula $Def_s^{\mathcal{S}}$. Then by Lemma 4.6 there is some $S'_s \subseteq (T \cup \{s\})$ (also $S'_s \subseteq S$) with $s \in S'_s$ such that $S'_s \in \mathcal{S}$ for each $s \in S$. We know from above that $C_{\mathcal{S}}(S'_{s_1} \cup S'_{s_2}) \subseteq S$ for any $s_1, s_2 \in S$.

As $\mathcal{S}$ is comp-closed and therefore $C_{\mathcal{S}}(S'_{s_1} \cup S'_{s_2}) \in \mathcal{S}$ also $S \in \mathcal{S}$.

□

Having shown that comp-closed extension-sets containing the intersection of all its elements are realizable in general, we conclude the topic of realizability under the complete semantics with a final example. In the examples seen so far the AFs obtained from Construction 4.5 had no attacks between the initial arguments, i.e. $\forall a, b \in \mathcal{S} : (a, b) \in Pairs_{\mathcal{S}}$. The following example will have some of these attacks.

Example 4.12. Consider the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{x\}, \{y\}, \{x, y\}, \{a, b, c\}\}$. The CNF-defense-formulas are

- $CDef_a^{\mathcal{S}} = CDef_b^{\mathcal{S}} = CDef_x^{\mathcal{S}} = CDef_y^{\mathcal{S}} = \{\}$ and
- $CDef_c^{\mathcal{S}} = \{\{a\}, \{b\}\}$,

and the CNF-completion-formulas are

- $CCom_a^{\mathcal{S}} = CCom_b^{\mathcal{S}} = CCom_x^{\mathcal{S}} = CCom_y^{\mathcal{S}} = \{\}$ and
- $CCom_c^{\mathcal{S}} = \{\{a\}, \{b\}\}$.

The framework $F_{\mathcal{S}}^{comp}$ obtained from Construction 4.5 is depicted in Figure 4.10. It holds that $comp(F_{\mathcal{S}}^{comp}) = \mathcal{S}$. One can see that the defense- and completion-formulas coincide for all $s \in Args_{\mathcal{S}}$ and that the arguments $\beta_{c,\{a\}}$ and $\beta_{c,\{b\}}$ and all in- and outgoing attacks could be omitted, because the arguments $\alpha_{c,\{a\}}$ and $\alpha_{c,\{b\}}$ already fill their role of enforcing c to be in an extension containing a and b.

Example 4.12 also shows a downside of Construction 4.5, as the AF $F_{\mathcal{S}}^{comp}$ we get from the construction contains many extra arguments, while $\mathcal{S}$ could be realized under the complete semantics by a much more compact framework. Construction 4.5 is defined generally in order to realize any comp-closed extension-set, therefore the size of a constructed AF can be much bigger than necessary. In Section 5.2 we will see a much smaller AF realizing the extension-set $\mathcal{S}$ from Example 4.12.

Corollary 4.13. *Each extension-set $\mathcal{S}$ which is adm-realizable is also comp-realizable.*

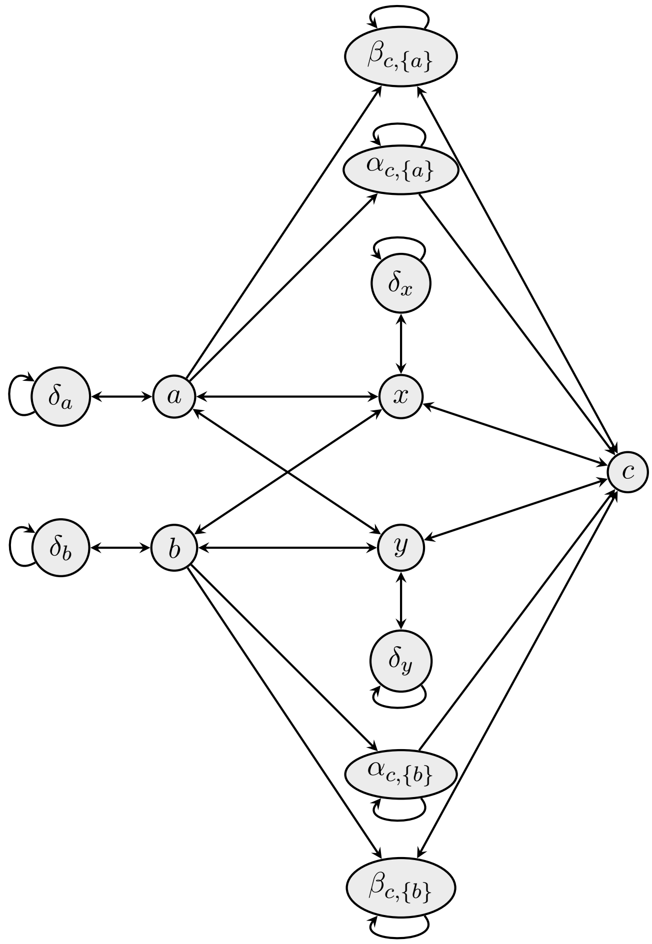

Figure 4.10: AF $F_{\mathcal{S}}^{comp}$ realizing $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{x\}, \{y\}, \{x, y\}, \{a, b, c\}\}$ under the complete semantics

Proof. As each adm-closed set is comp-closed by Lemma 3.10 and for each $\mathcal{S}$ containing $\emptyset$ it holds that $\bigcap_{S \in \mathcal{S}} S \in \mathcal{S}$, this follows immediately from Proposition 4.8 and Proposition 4.12. □

We have seen that the necessary conditions from Chapter 3 coincide with the sufficient conditions in terms of realizability given in this chapter. Table 4.1 summarizes, which of the extension-sets defined in Example 3.1 are realizable with the semantics we consider. An entry in line σ and column $\mathcal{S}$ has to be interpreted as follows:

- A checkmark ✓ indicates that $\mathcal{S}$ is realizable under the semantics σ.
- In addition, a number of the form $x.y$ indicates that $\mathcal{S}$ is realizable under the semantics σ and the AF doing so is depicted in Figure $x.y$.

	$\mathcal{S}_0$	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$	$\mathcal{S}_4$	$\mathcal{S}_5$	$\mathcal{S}_6$	$\mathcal{S}_7$
cf	-	4.1	-	-	-	-	-	-
naive	-	-	3.1	-	-	-	-	-
stage	-	-	3.1	3.2	-	-	-	-
stb	4.2	-	3.1	3.2	-	-	-	-
adm	-	4.1	(✓)	(✓)	(✓)	(4.5)	-	-
pref	-	-	3.1	3.2	3.3	-	-	-
sem	-	-	3.1	3.2	3.3	-	-	-
comp	-	✓	(✓)	3.2	(✓)	(✓)	4.8	-

Table 4.1: Realizability of the extension-sets from Example 3.1 under the semantics of our interest

- If the number or the checkmark is in brackets, then not $\mathcal{S}$ but $\mathcal{S} \cup \{\emptyset\}$ is realizable under the semantics σ.
- A "–" indicates that $\mathcal{S}$ is not realizable under the semantics σ.

4.2 Strict Realizability

In Section 4.1, we have investigated which sets of extensions can be realized by argumentation frameworks under a particular semantics σ. In other words, given a collection $\mathcal{S}$ of sets of arguments, we clarified whether there exists an AF F such that $\sigma(F) = \mathcal{S}$. The following sharpening of such a concept is natural: does there exist an AF $F = (\mathit{Args}_{\mathcal{S}}, R)$ such that $\sigma(F) = \mathcal{S}$, i.e. can we find an AF without additional arguments which do not appear in the desired extensions. We call a set of extensions *strictly realizable* under the semantics σ if such an AF exists.

We make this idea formal:

Definition 4.4. Given a semantics σ, an extension-set $\mathcal{S}$ is called *strictly σ-realizable* if there exists an AF $F = (\mathit{Args}_{\mathcal{S}}, R)$ such that $\sigma(F) = \mathcal{S}$. $\mathcal{S}$ is then *strictly realized* by F under σ.

We will see that for the conflict-free and naive semantics general and strict realizability coincide. For the other semantics we deal with in this thesis this does not hold. We will show why this is the case by giving examples of extension-sets, which are proven to be generally realizable but are evidentially not strictly realizable.

We begin with the conflict-free semantics.

Proposition 4.14. *Each non-empty, downward-closed, and tight extension-set $\mathcal{S}$ is strictly cf-realized by $F^{cf}_{\mathcal{S}}$.*

Proof. We know from Proposition 4.1 that each non-empty, downward-closed and tight extension-set $\mathcal{S}$ is *cf*-realizable by $F_{\mathcal{S}}^{cf}$. As Construction 4.1 does not introduce additional arguments other than $Args_{\mathcal{S}}$, $\mathcal{S}$ is strictly *cf*-realizable. □

Corollary 4.15. *Each extension-set $\mathcal{S}$ which is cf-realizable, is also strictly cf-realizable.*

As an example, the AF in Figure 4.1 strictly realizes the extension-set $\mathcal{S}_1 = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,c\}\}$ under the conflict-free semantics.

We continue with the naive semantics.

Proposition 4.16. *Each incomparable and non-empty extension-set $\mathcal{S}$, where $dcl(\mathcal{S})$ is tight, is strictly naive-realized by $F_{\mathcal{S}}^{cf}$.*

Proof. We know from Proposition 4.3 that each incomparable and non-empty extension-set $\mathcal{S}$, where $dcl(\mathcal{S})$ is tight, is *naive*-realizable by $F_{\mathcal{S}}^{cf}$. As Construction 4.1 does not introduce additional arguments other than $Args_{\mathcal{S}}$, $\mathcal{S}$ is strictly *naive*-realizable. □

Corollary 4.17. *Each extension-set $\mathcal{S}$ which is naive-realizable, is also strictly naive-realizable.*

Figure 3.1 shows the AF $F_{\mathcal{S}_2}^{cf}$, which strictly realizes the extension-set $\mathcal{S}_2$ from Example 3.1 under the naive semantics.

This correspondence between general and strict realizability does not hold for the other semantics discussed in this thesis. We show this by giving examples, where extension-sets fulfilling the conditions for general realizability are not strictly realizable.

We begin with the stable semantics.

Proposition 4.18. *There exists an incomparable and tight extension-set $\mathcal{S}$, which is not strictly stb-realizable.*

Proof. Consider the extension-set $\mathcal{S} = \{\{a,b,c\}, \{a,b,c'\}, \{a,b',c\}, \{a',b,c\}, \{a,b',c'\}, \{a',b,c'\}, \{a',b',c\}\}$. It is easy to verify that $\mathcal{S}$ is incomparable and tight. Hence, by Proposition 4.4, it follows that $\mathcal{S}$ is *stb*-realizable. However the AF provided by Construction 4.2 makes use of an argument not in $Args_{\mathcal{S}} = \{a,b,c,a',b',c'\}$. Figure 4.11 shows $F_{\mathcal{S}}^{st}$ with the additional argument $\bar{E}$.

We now show that there is no AF $F = (Args_{\mathcal{S}}, R)$ such that $stb(F) = \mathcal{S}$. First, given that the sets in $\mathcal{S}$ must be conflict-free the only possible attacks in R are (a,a'), (a',a), (b,b'), (b',b), (c,c'), (c',c). We next argue that all of them must be in R. As $\{a,b,c\} \in stb(F)$ it attacks a' and since (b,a') and (c,a') are in $Pairs_{\mathcal{S}}$ the only chance to do so is by $(a,a') \in R$ and similar as $\{a',b,c\} \in stb(F)$ it attacks a and since (b,a) and (c,a) are in $Pairs_{\mathcal{S}}$ the only chance to do so is $(a',a) \in R$. By symmetry we obtain $\{(b,b'), (b',b), (c,c'), (c',c)\} \subseteq R$. Hence we obtain the only possible attack relation $R = \{(a,a'), (a',a), (b,b'), (b',b), (c,c'), (c',c)\}$. However, for the resulting framework $F = (Args_{\mathcal{S}}, R)$, we have that $\{a',b',c'\} \in stb(F)$, but $\{a',b',c'\} \notin \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *stb*-realizable. □

We use the same example, with slightly different argumentation though, for stage semantics.

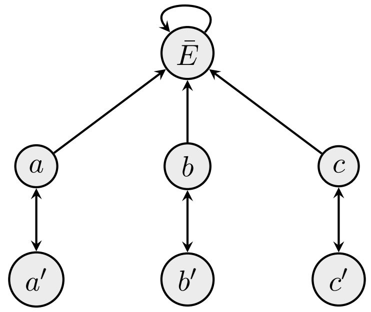

Figure 4.11: AF $F^{st}_{\mathcal{S}}$ where $\mathcal{S} = \{\{a,b,c\}, \{a,b,c'\}, \{a,b',c\}, \{a,b',c'\}, \{a',b,c\}, \{a',b,c'\}, \{a',b',c\}\}$

Proposition 4.19. *There exists a non-empty, incomparable and tight extension-set $\mathcal{S}$, which is not strictly stage-realizable.*

Proof. Again consider the extension-set $\mathcal{S} = \{\{a,b,c\}, \{a,b,c'\}, \{a,b',c\}, \{a',b,c\}, \{a,b',c'\}, \{a',b,c\}, \{a',b,c'\}, \{a',b',c\}\}$. It is easy to verify that $\mathcal{S}$ is not only incomparable and tight, but also non-empty. Hence, by Proposition 4.5, it follows that $\mathcal{S}$ is *stage*-realizable. We have seen in Proposition 4.18 that the AF provided by Construction 4.2 (see Figure 4.11) makes use of an additional argument.

We now show that there is no AF $F = (Args_{\mathcal{S}}, R)$ such that $stage(F) = \mathcal{S}$. Again the only possible attacks in R are (a,a'), (a',a), (b,b'), (b',b), (c,c'), (c',c). We next argue that all of them must be in R. As $\{a,b,c\} \in stage(F)$ and $stage(F) \subseteq naive(F)$ either $(a,a') \in R$ or $(a',a) \in R$. Consider $(a,a') \notin R$ then $\{a',b,c\}^+ \supset \{a,b,c\}^+$, hence $\{a,b,c\} \notin stage(F)$, a contradiction. The same holds for pairs (b,b') and (c,c'). Thus again we obtain the attack-relation $R = \{(a,a'),(a',a),(b,b'),(b',b),(c,c'),(c',c)\}$ and the AF $F = (Args_{\mathcal{S}}, R)$ depicted in Figure 4.12. We have that $\{a',b',c'\} \in stage(F)$, but $\{a',b',c'\} \notin \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *stage*-realizable. □

Also for the admissible semantics, general and strict realizability do not coincide:

Proposition 4.20. *There exists an adm-closed extension-set $\mathcal{S}$ containing $\emptyset$, which is not strictly adm-realizable.*

Proof. Consider the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{a,b\}\}$. It is easy to see that $\mathcal{S}$ is adm-closed, cf. Definition 3.7. Indeed the AF $F = (\{a,b,c\}, \{(a,c),(c,b)\})$ (see Figure 4.13) realizes

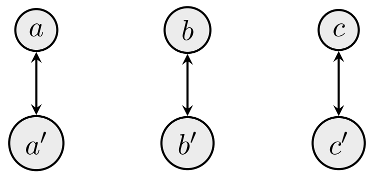

Figure 4.12: AF F used in Proposition 4.18 and Proposition 4.19

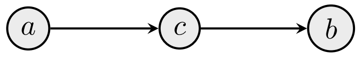

Figure 4.13: AF F having $\{\emptyset, \{a\}, \{a, b\}\}$ as its admissible sets

$\mathcal{S}$ under the admissible semantics, i.e. $adm(F) = \mathcal{S}$. However, there does not exist an AF $F' = (A, R)$ with $adm(F') = \mathcal{S}$ and $A = \{a, b\}$, since by $\{a, b\} \in \mathcal{S}$ there cannot be any attack between a and b nor any self attacks in F'. But then $adm(F') = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$ and by that $adm(F') \neq \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *adm*-realizable. □

We can give the following sufficient conditions for extension-sets to be strictly realizable under the admissible semantics.

Proposition 4.21. *Any extension-set which is non-empty, downward-closed, and tight is strictly adm-realizable.*

Proof. Proposition 4.1 shows that for each non-empty, downward-closed and tight extension-set $\mathcal{S}$, it holds that $cf(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. As $F^{cf}_{\mathcal{S}}$ is symmetric and $R_{F^{cf}_{\mathcal{S}}}$ is irreflexive, $adm(F^{cf}_{\mathcal{S}}) = \mathcal{S}$ by Proposition 2.15. As Construction 4.1 does not introduce additional arguments other than $Args_{\mathcal{S}}$, $\mathcal{S}$ is strictly *adm*-realizable. □

Note that there are extension-sets not fulfilling the properties given in Proposition 4.21 which are strictly realizable under the admissible semantics as well. For example consider the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{c\}, \{a, b\}\}$. $\mathcal{S}$ is not downward-closed as $\{b\} \notin \mathcal{S}$, but the AF $F = (\{a, b, c\}, \{(a, b), (b, a), (b, c)\})$ strictly realizes $\mathcal{S}$ under the admissible semantics, i.e. $adm(F) = \mathcal{S}$ and $A_F = Args_{\mathcal{S}}$.

Proposition 4.22. *There exists a non-empty and pref-closed extension-set $\mathcal{S}$ which is neither strictly pref-realizable nor strictly sem-realizable.*

Proof. Consider the extension-set $\mathcal{S} = \{\{a, b\}, \{a, c, e\}, \{b, d, e\}\}$. It is easy to see that $\mathcal{S}$ is pref-closed, cf. Definition 3.8. Figure 3.3 shows an AF realizing $\mathcal{S}$ under the semi-stable and preferred semantics. However, that AF contains the additional arguments a', b' and f not occurring in $Args_{\mathcal{S}}$. Suppose there exists an AF $F = (Args_{\mathcal{S}}, R)$ such that $\sigma(F) = \mathcal{S}$ for $\sigma = \{pref, sem\}$. Since $\{a, c, e\}, \{b, d, e\} \in \mathcal{S}$, it is clear that R must not contain an edge involving e. But then, e is contained in each extension $E \in \sigma(F)$ (in particular, for the case of semi-stable extensions, since e is not attacked in such an F). It follows that $\sigma(F) \neq \mathcal{S}$, and therefore $\mathcal{S}$ is neither strictly *pref*-realizable nor strictly *sem*-realizable. □

In other words Proposition 4.22 shows that being non-empty and pref-closed are not sufficient conditions for an extension-set to be *pref*-realizable or *sem*-realizable. Proposition 4.23 shows that even stricter conditions (non-empty, incomparable, and tight) are not sufficient in order to be realized under preferred semantics as well as under semi-stable semantics.

Proposition 4.23. *There exists a non-empty, incomparable, and tight extension-set $\mathcal{S}$ which is neither strictly pref-realizable nor strictly sem-realizable.*

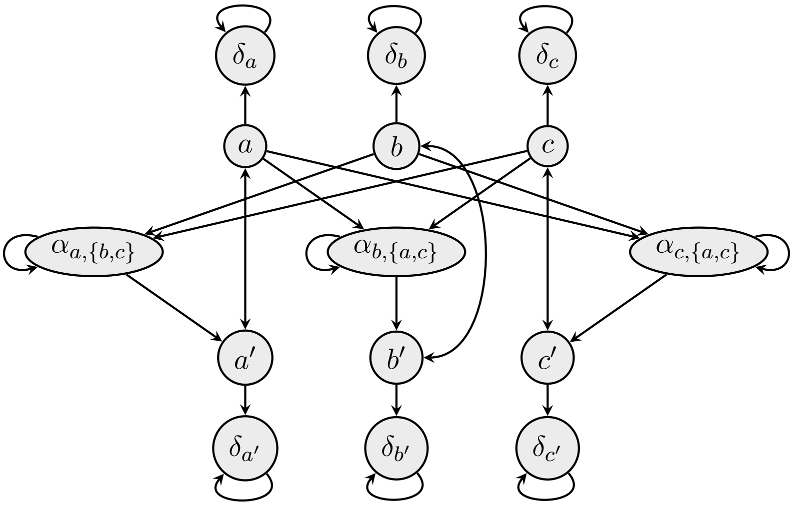

Figure 4.14: AF $F_{\mathcal{S}}^{sem}$ where $\mathcal{S} = \{\{a, b, c\}, \{a, b, c'\}, \{a, b', c\}, \{a, b', c'\}, \{a', b, c\}, \{a', b, c'\}, \{a', b', c\}\}$

Proof. As before consider the extension-set $\mathcal{S} = \{\{a, b, c\}, \{a, b, c'\}, \{a, b', c\}, \{a, b', c'\}, \{a', b, c\}, \{a', b, c'\}, \{a', b', c\}\}$. It is easy to verify that $\mathcal{S}$ is non-empty, incomparable and tight. Hence, by Lemma 3.5 and Proposition 4.9, $\mathcal{S}$ is *pref*-realizable and further by Proposition 4.11 it is *sem*-realizable. The AF $F_{\mathcal{S}}^{sem}$ realizing $\mathcal{S}$ under the preferred and semi-stable semantics is depicted in Figure 4.14. Note that the arguments $\{\delta_x \mid x \in Args_{\mathcal{S}}\}$ and their corresponding attacks are not necessary in order to realize $\mathcal{S}$ under the preferred semantics. However in both cases the AF makes use of additional arguments. We now show that there is no AF $F = (Args_{\mathcal{S}}, R)$ such that $pref(F) = \mathcal{S}$ or $sem(F) = \mathcal{S}$.

Assume there is such an AF $F = (Args_{\mathcal{S}}, R)$. First, given that the sets in $\mathcal{S}$ must be conflict-free the only possible attacks in R are (a, a'), (a', a), (b, b'), (b', b), (c, c'), (c', c). We next argue that all of them must be in R.

- *pref*: As $\{a', b, c\} \in pref(F)$ it must be $\subseteq$-maximal. Consider the argument a. There must be a conflict between $\{a', b, c\}$ and a in order to preserve $\subseteq$-maximality. The only way to do so is $(a, a') \in R$ and/or $(a', a) \in R$. Assume $(a, a') \in R$ then for $\{a', b, c\}$ to stay admissible, also $(a', a) \in R$. On the other hand assume $(a', a) \in R$ then for $\{a, b', c'\} \in \mathcal{S}$ to stay admissible, also $(a', a) \in R$. The same holds symmetrically for b and c, so F is the AF in Figure 4.12. But then $\{a', b', c'\} \in pref(F)$, but $\{a', b', c'\} \notin \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *pref*-realizable.

- *sem*: As $\{a, b, c\} \in \mathcal{S}$, but $\{a', b', c'\} \notin \mathcal{S}$ one of the following must hold: $\{a', b', c'\} \notin adm(F)$ or $\{a, b, c\}^+ \supset \{a', b', c'\}^+$. Considering the possible attacks both can only be the case if $\{a, b, c\} \mapsto^R \{a', b', c'\}$. Without loss of generality assume $(a, a') \in R$. Then for $\{a', b, c\} \in \mathcal{S}$ to be admissible also $(a', a) \in R$. The same holds symmetrically for b

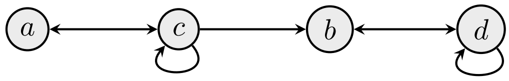

Figure 4.15: AF F realizing $\mathcal{S} = \{\emptyset, \{a\}, \{a, b\}\}$ under complete semantics

and c, so F is the AF in Figure 4.12. But then $\{a', b', c'\} \in \mathit{sem}(F)$, but $\{a', b', c'\} \notin \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *sem*-realizable.

□

Proposition 4.24. *There exists a comp-closed extension-set* $\mathcal{S}$ *where* $\bigcap_{S \in \mathcal{S}} S \in \mathcal{S}$*, which is not strictly comp-realizable.*

Proof. Consider $\mathcal{S} = \{\emptyset, \{a\}, \{a, b\}\}$. It is easy to see that $\mathcal{S}$ is comp-closed, cf. Definition 3.10, and that $\bigcap_{S \in \mathcal{S}} S = \emptyset \in \mathcal{S}$. Indeed $\mathcal{S}$ is *comp*-realizable, namely for instance by the AF $F = (\{a, b, c, d\}, \{(a, c), (c, a), (c, c), (c, b), (b, d), (d, b), (d, d)\})$ (see Figure 4.15), where $\mathit{comp}(F) = \mathcal{S}$. However, there does not exist an $F' = (A, R)$ with $\sigma(F') = \mathcal{S}$ and $A = \{a, b\}$, since by $\{a, b\} \in \mathcal{S}$ there cannot be any attack between a and b nor any self attacks in F'. But then $\mathit{comp}(F') = \{\{a, b\}\}$ and obviously $\mathit{comp}(F') \neq \mathcal{S}$. Therefore $\mathcal{S}$ is not strictly *comp*-realizable. □

In Propositions 4.18, 4.19, 4.20, 4.22 and 4.24 we have shown that for stable, stage, admissible, preferred, semi-stable and complete semantics general and strict realizability do not coincide. First investigations indicate that finding exact characterizations for extension-sets to be strictly realizable is a non-trivial task. Having some ideas in mind, we leave this to future work.

CHAPTER 5

Signatures

We are now ready to give our main results. Assembling the results of Chapter 3 and Chapter 4 we can formally define the signature of a semantics and give a full characterization of the signatures of the semantics of our interest. Further we are going to relate these signatures to each other and by that compare the levels of expressiveness of the semantics. Finally we will show the connections to the recent work on intertranslatability [38, 39, 57], discussing similarities but also differences.

5.1 Signatures of Argumentation Semantics

In Chapter 3 we have dealt with properties of semantics which hold for every argumentation framework. In particular we have given characteristics for the sets of extensions provided by certain semantics which describe the shape of these extension-sets no matter how the underlying AF looks like.

Then, in Chapter 4 we have examined, which properties an extension-set has to fulfill in order to guarantee that there is some AF that realizes this extension-set under a particular semantics, i.e. the extensions under the semantics coincide with the extension-set.

The combination of these two aspects leads us to the definition of a *signature*. The signature of a semantics σ is the set of extension-sets which can be realized by some AF under the semantics σ.

Definition 5.1. The signature Σ_σ of a semantics σ is defined as

$$\Sigma_\sigma = \{\sigma(F) \mid F \in AF_{\mathcal{A}}\}\,.$$

So consider some semantics σ. If, given an arbitrary AF, the set of extensions under σ fulfills a certain property, then this property is just a necessary condition for an extension-set to be in the signature Σ_σ. On the other hand, if all extension-sets fulfilling a certain property can be realized under the semantics σ, then this property is a sufficient condition for an extension-set to be in the signature Σ_σ. If the necessary and sufficient conditions coincide, we have the description of which extension-sets the signature is composed of. As this is the case for the propositions given in Chapters 3 and 4 we can give the signatures for the semantics of our interest.

Theorem 5.1. *The signatures for the considered semantics of interest are given by the following collections of extension sets.*

$$\Sigma_{cf} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is downward-closed and tight}\} \tag{5.1}$$
$$\Sigma_{naive} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is incomparable and } dcl(\mathcal{S}) \textit{ is tight}\} \tag{5.2}$$
$$\Sigma_{stb} = \{\mathcal{S} \mid \mathcal{S} \textit{ is incomparable and tight}\} \tag{5.3}$$
$$\Sigma_{stage} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is incomparable and tight}\} \tag{5.4}$$
$$\Sigma_{adm} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is adm-closed and contains } \emptyset\} \tag{5.5}$$
$$\Sigma_{pref} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is pref-closed}\} \tag{5.6}$$
$$\Sigma_{sem} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is pref-closed}\} \tag{5.7}$$
$$\Sigma_{comp} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is comp-closed and contains } \bigcap_{S \in \mathcal{S}} S\} \tag{5.8}$$

Proof. (5.1): As for each AF F it holds that $cf(F)$ is non-empty, downward closed and tight by Proposition 3.11, $\Sigma_{cf} \subseteq \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is downward-closed and tight}\}$ holds. On the other hand since for each non-empty, downward closed and tight extension-set $\mathcal{S}$ there is some AF F with $cf(F) = \mathcal{S}$ by Proposition 4.1 it holds that $\Sigma_{cf} \supseteq \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is downward-closed and tight}\}$. Hence the characterization of Σ_{cf} follows. We obtain the signatures of the remaining semantics in the same manner.
(5.2): Σ_{naive} follows from Proposition 3.13 and Proposition 4.3.
(5.3): Σ_{stb} follows from Proposition 3.15 and Proposition 4.4.
(5.4): Σ_{stage} follows from Proposition 3.14 and Proposition 4.5.
(5.5): Σ_{adm} follows from Proposition 3.16 and Proposition 4.8.
(5.6): Σ_{pref} follows from Proposition 3.17 and Proposition 4.9.
(5.7): Σ_{sem} follows from Proposition 3.19 and Proposition 4.11.
(5.8): Σ_{comp} follows from Proposition 3.20 and Proposition 4.12. □

In Section 4.2 we have gone one step further and examined, which extension-sets can be realized under a certain semantics without introducing additional arguments. This gives rise to the introduction of the *strict signature*.

Definition 5.2. The strict signature Σ^s_σ of a semantics σ is defined as

$$\Sigma^s_\sigma = \left\{\sigma(F) \mid F \in AF_{\mathcal{A}} \text{ with } A_F = Args_{\sigma(F)}\right\}.$$

In words, some extension-set $\mathcal{S}$ is in the strict signature of some semantics σ, i.e. $\mathcal{S} \in \Sigma^s_\sigma$, if there is some AF F, where the arguments A_F coincide with the arguments contained in $\mathcal{S}$, i.e. $A_F = Args_{\mathcal{S}}$ and $\sigma(F) = \mathcal{S}$. We give the characterizations for the strict signatures we know about so far.

Theorem 5.2. *The strict signatures for the considered semantics of interest are given by the following collections of extension sets.*

$$\Sigma^s_{cf} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is downward-closed and tight}\} \tag{5.9}$$

$$\Sigma^s_{naive} = \{\mathcal{S} \neq \emptyset \mid \mathcal{S} \textit{ is incomparable and } dcl(\mathcal{S}) \textit{ is tight}\} \tag{5.10}$$

Proof. (5.9) follows from Proposition 3.11 and Proposition 4.1 and the fact that Construction 4.1 does not introduce additional arguments.

(5.10) follows from Proposition 3.13 and Proposition 4.3 and the fact that Construction 4.1 does not introduce additional arguments. □

It is easy to see that for any semantics σ it holds that $\Sigma^s_\sigma \subseteq \Sigma_\sigma$. For $\sigma \in \{stage, stb, adm, pref, sem, comp\}$ however, $\Sigma^s_\sigma \supseteq \Sigma_\sigma$ does not hold and therefore $\Sigma^s_\sigma \subset \Sigma_\sigma$. We will come to that in Theorems 5.8, 5.9, 5.10, 5.11 and 5.13.

Let us now address the topic of whether the empty set of arguments should be an extension or not. As we have seen in Section 2.2, given an AF F, the empty set of arguments is always a conflict-free set in F as well as an admissible set in F. On the other hand $\emptyset$ is a naive, preferred, stable, stage and semi-stable extension in F if and only if F possesses no other extension under the given semantics. Only the complete semantics behaves a little more involved. Here $\emptyset$ is a complete extension if it is the grounded extension, which is the case if there is no argument without ingoing attacks.

Accepting the empty set of arguments as an extension can be seen as leaving open the opportunity that none of the arguments is acceptable. We have seen that the different semantics decide about that more or less independently from the given AF. In order to take a look at the expressiveness of semantics without caring about whether the semantics accepts the empty set of arguments as an extension, we introduce the $\emptyset$-extended signature.

Definition 5.3. The $\emptyset$-extended signature $\Sigma^\emptyset_\sigma$ of a semantics σ is defined as

$$\Sigma_\sigma = \{\sigma(F) \cup \{\emptyset\} \mid F \in AF_\mathcal{A}\}.$$

The $\emptyset$-extended signature of a semantics σ is the set of extension-sets, which can be realized by some AF under the semantics σ, with each of them extended by the empty extension. We will see in Section 5.2 that considering the $\emptyset$-extended signatures makes a difference in the relations between the semantics.

5.2 Comparing the Levels of Disagreement

The signatures given in Theorem 5.1 describe the sets of extensions which can be the outcome of applying a certain semantics to an arbitrary AF. In other words a signature describes the levels of disagreement that can be expressed by a semantics. In the following we are going to compare the semantics of our interest in terms of their capabilities of expressiveness.

Theorem 5.3. *The expressiveness of conflict-free, admissible, and complete semantics is related in the following way:*

$$\Sigma_{cf} \subset \Sigma_{adm} \subset \Sigma_{comp}.$$

Proof. Every downward-closed and tight extension-set $\mathcal{S}$ is adm-closed by Lemma 3.4. As for each $\mathcal{S} \in \Sigma_{cf}$ it holds that $\mathcal{S} \neq \emptyset$ and therefore $\emptyset \in \mathcal{S}$ by $\mathcal{S}$ being downward-closed, $\Sigma_{cf} \subseteq \Sigma_{adm}$ holds.

As $\mathcal{S}_5 = \{\{a\}, \{b, c\}, \{a, c, d\}\}$ from Example 3.1 is neither downward-closed nor tight, but adm-closed though (see Examples 3.3 and 3.5), $\mathcal{S}_5 \in \Sigma_{adm}$, but $\mathcal{S}_5 \notin \Sigma_{cf}$, therefore $\Sigma_{cf} \not\supseteq \Sigma_{adm}$. Hence $\Sigma_{cf} \subset \Sigma_{adm}$.

By Lemma 3.10 an adm-closed extension-set is also comp-closed. As each $\mathcal{S} \in \Sigma_{adm}$ contains $\emptyset$, $\bigcap_{S \in \mathcal{S}} S = \emptyset$. Therefore $\bigcap_{S \in \mathcal{S}} S \in \mathcal{S}$, hence $\Sigma_{adm} \subseteq \Sigma_{comp}$.

In order to show that $\Sigma_{adm} \not\supseteq \Sigma_{comp}$, consider the comp-closed (cf. Example 3.8) extension-set $\mathcal{S}'_6 = (\mathcal{S}_6 \cup \{\emptyset\})$ where $\mathcal{S}_6 = \{\{a\}, \{b\}, \{c\}, \{a, b, c\}\}$ from Example 3.1. It holds that $\mathcal{S}'_6 \in \Sigma_{comp}$, it is realized under the complete semantics by the AF in Figure 4.8. As $\mathcal{S}_6$ is not adm-closed (cf. Example 3.5), $\mathcal{S}_6 \notin \Sigma_{adm}$, therefore $\Sigma_{adm} \subset \Sigma_{comp}$. □

Theorem 5.4. *The expressiveness of naive, stage, preferred, semi-stable, and complete semantics is related in the following way:*

$$\Sigma_{naive} \subset \Sigma_{stage} \subset \Sigma_{pref} = \Sigma_{sem}.$$

Proof. Since for each extension-set $\mathcal{S}$ where $dcl(\mathcal{S})$ is tight, it holds that $\mathcal{S}$ is tight too by Corollary 3.2, it follows that $\Sigma_{naive} \subseteq \Sigma_{stage}$.

Now consider $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$ from Example 3.1. Note that $dcl(\mathcal{S}_3)$ is not tight, therefore $\mathcal{S}_3 \notin \Sigma_{naive}$. $\mathcal{S}_3$ is tight (cf. Example 3.3), hence $\mathcal{S}_3 \in \Sigma_{stage}$, e.g. realized under the stage semantics by the AF in Figure 3.2. So $\Sigma_{naive} \not\supseteq \Sigma_{stage}$ and therefore $\Sigma_{naive} \subset \Sigma_{stage}$.

We know by Lemma 3.5 that each incomparable and tight extension-set is also pref-closed. Therefore $\Sigma_{stage} \subseteq \Sigma_{pref}$.

As $\mathcal{S}_4 = \{\{a, b\}, \{a, c, e\}, \{b, d, e\}\}$ from Example 3.1 is not tight, $\mathcal{S}_4 \notin \Sigma_{stage}$. But since $\mathcal{S}_4$ is pref-closed (see Example 3.6) it holds that $\mathcal{S}_4 \in \Sigma_{pref}$. The AF in Figure 3.3 realizes $\mathcal{S}_4$ under the preferred semantics. We follow that $\Sigma_{stage} \not\supseteq \Sigma_{pref}$ and therefore $\Sigma_{stage} \subset \Sigma_{pref}$.

$\Sigma_{pref} = \Sigma_{sem}$ holds as the characterizations coincide according to Theorem 5.1. □

Theorem 5.5. *The expressiveness of stable and stage semantics differs only in the empty set of extensions:*

$$\Sigma_{stage} = (\Sigma_{stb} \setminus \{\emptyset\}).$$

Proof. This follows immediately from the characterizations of Σ_{stage} and Σ_{stb} in Theorem 5.1. □

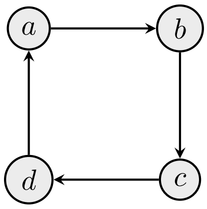

Figure 5.1: AF strictly realizing $\mathcal{S} = \{\emptyset, \{a, c\}, \{b, d\}\}$ under the admissible semantics

Theorem 5.6. *The expressiveness of conflict-free and naive semantics is related in the following way:*

$$\Sigma_{cf} = \{dcl(\mathcal{S}) \mid \mathcal{S} \in \Sigma_{naive}\}.$$

Proof. This is easy to see as for each $\mathcal{S} \in \Sigma_{naive}$, $dcl(\mathcal{S})$ is tight and of course downward-closed. □

Theorem 5.7. *The expressiveness of preferred and admissible semantics is related in the following way:*

$$\{\mathcal{S} \cup \{\emptyset\} \mid \mathcal{S} \in \Sigma_{pref}\} \subset \Sigma_{adm}.$$

Proof. As we know from Lemma 3.9 for a pref-closed extension-set $\mathcal{S}$ it holds that $\mathcal{S} \cup \{\emptyset\}$ is adm-closed. Therefore $\{\mathcal{S} \cup \{\emptyset\} \mid \mathcal{S} \in \Sigma_{pref}\} \subseteq \Sigma_{adm}$.

To show that $\{\mathcal{S} \cup \{\emptyset\} \mid \mathcal{S} \in \Sigma_{pref}\} \supseteq \Sigma_{adm}$ cannot hold, consider the extension-set $\mathcal{S} = \{\{a\}, \{a, b\}\}$. Of course $(\mathcal{S} \cup \{\emptyset\}) \in \Sigma_{adm}$, but as $\mathcal{S}$ is not incomparable, it is surely not pref-closed and therefore $(\mathcal{S} \cup \{\emptyset\}) \notin \{\mathcal{S} \cup \{\emptyset\} \mid \mathcal{S} \in \Sigma_{pref}\}$. It follows that $\{\mathcal{S} \cup \{\emptyset\} \mid \mathcal{S} \in \Sigma_{pref}\} \subset \Sigma_{adm}$. □

Theorem 5.8. *The following relations hold:*

$$\Sigma_{cf} \subset \Sigma^s_{adm} \subset \Sigma_{adm}$$

Proof. $\Sigma_{cf} \subseteq \Sigma^s_{adm}$ follows from Proposition 4.21. To show that $\Sigma_{cf} \not\supseteq \Sigma^s_{adm}$, consider the extension-set $\mathcal{S} = \{\emptyset, \{a, c\}, \{b, d\}\}$. $\mathcal{S}$ is not downward-closed, hence $\mathcal{S} \notin \Sigma_{cf}$. As under the admissible semantics $\mathcal{S}$ is strictly realized by the AF in Figure 5.1 it holds that $\mathcal{S} \in \Sigma^s_{adm}$ and therefore $\Sigma_{cf} \subset \Sigma^s_{adm}$.

While $\Sigma^s_{adm} \subseteq \Sigma_{adm}$ trivially holds, we have seen $\Sigma^s_{adm} \not\supseteq \Sigma_{adm}$ in Proposition 4.20. □

Theorem 5.9. *The following relations hold:*

$$\Sigma_{naive} \subset \Sigma^s_{stage} \subset \Sigma_{stage}$$

Proof. First consider some extension-set $\mathcal{S} \in \Sigma_{naive}$. $\mathcal{S}$ is incomparable, non-empty and $dcl(\mathcal{S})$ is tight. By Proposition 4.3 $naive(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. As $F^{cf}_{\mathcal{S}}$ is symmetric and $R_{F^{cf}_{\mathcal{S}}}$ is irreflexive, $stage(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. As $A_{F^{cf}_{\mathcal{S}}} = Args_{\mathcal{S}}$, i.e. $F^{cf}_{\mathcal{S}}$ does not use additional arguments, it follows that $\mathcal{S} \in \Sigma^{s}_{stage}$ and therefore $\Sigma_{naive} \subseteq \Sigma^{s}_{stage}$.

In order to show that $\Sigma_{naive} \not\supseteq \Sigma^{s}_{stage}$ consider extension-set $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$ from Example 3.1. $dcl(\mathcal{S}_3)$ is not tight, hence $\mathcal{S}_3 \notin \Sigma_{naive}$. Since $\mathcal{S}_3$ is realized by the AF in Figure 3.2, $\mathcal{S}_3 \in \Sigma_{stage}$ and therefore $\Sigma_{naive} \subset \Sigma^{s}_{stage}$.

While $\Sigma^{s}_{stage} \subseteq \Sigma_{stage}$ trivially holds, we have seen $\Sigma^{s}_{stage} \not\supseteq \Sigma_{stage}$ in Proposition 4.19. □

Theorem 5.10. *The following relations hold:*

$$\Sigma_{naive} \subset \Sigma^{s}_{stb} \subset \Sigma_{stb}$$

Proof. Consider some extension-set $\mathcal{S} \in \Sigma_{naive}$. $\mathcal{S}$ is incomparable, non-empty and $dcl(\mathcal{S})$ is tight. By Proposition 4.3 $naive(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. As $F^{cf}_{\mathcal{S}}$ is symmetric and $R_{F^{cf}_{\mathcal{S}}}$ is irreflexive, $stb(F^{cf}_{\mathcal{S}}) = \mathcal{S}$. As $A_{F^{cf}_{\mathcal{S}}} = Args_{\mathcal{S}}$ it follows $\mathcal{S} \in \Sigma^{s}_{stb}$ and therefore $\Sigma_{naive} \subseteq \Sigma^{s}_{stb}$.

To show that $\Sigma_{naive} \not\supseteq \Sigma^{s}_{stb}$, again consider extension-set $\mathcal{S}_3 = \{\{a_1, b_2, b_3\}, \{a_2, b_1, b_3\}, \{a_3, b_1, b_2\}\}$ from Example 3.1. We know from Theorem 5.9 that $\mathcal{S}_3 \notin \Sigma_{naive}$. Since $\mathcal{S}_3$ is realized by the AF in Figure 3.2, $\mathcal{S}_3 \in \Sigma_{stb}$ and therefore $\Sigma_{naive} \subset \Sigma^{s}_{stb}$.

While $\Sigma^{s}_{stb} \subseteq \Sigma_{stb}$ trivially holds, we have seen $\Sigma^{s}_{stb} \not\supseteq \Sigma_{stb}$ in Proposition 4.18. □

Theorem 5.11. *The following relations hold for $\sigma \in \{sem, pref\}$:*

$$\Sigma^{s}_{\sigma} \subset \Sigma_{\sigma},$$

$$\Sigma_{stage} \not\subseteq \Sigma^{s}_{\sigma},$$

$$\Sigma_{stb} \not\subseteq \Sigma^{s}_{\sigma}.$$

Proof. First note that the signatures of preferred and semi-stable semantics, Σ_{pref} and Σ_{sem} coincide (cf. 5.1).

While $\Sigma^{s}_{\sigma} \subseteq \Sigma_{\sigma}$ trivially holds, it holds by Proposition 4.22 that there is a non-empty and pref-closed extension-set $\mathcal{S}$ which is neither strictly *pref*-realizable nor strictly *sem*-realizable, and therefore $\Sigma^{s}_{\sigma} \not\supseteq \Sigma_{\sigma}$ and hence $\Sigma^{s}_{\sigma} \subset \Sigma_{\sigma}$ for $\sigma \in \{sem, pref\}$.

By Proposition 4.23 there is a non-empty, incomparable and tight extension-set $\mathcal{S}$ which is neither strictly *pref*-realizable nor strictly *sem*-realizable. However for such an $\mathcal{S}$, there is some AF F, such that $stage(F) = \mathcal{S}$ and $stb(F) = \mathcal{S}$ (cf. Propositions 4.5 and 4.4). Therefore $\Sigma_{stage} \not\subseteq \Sigma^{s}_{\sigma}$ and $\Sigma_{stb} \not\subseteq \Sigma^{s}_{\sigma}$ for $\sigma \in \{sem, pref\}$. □

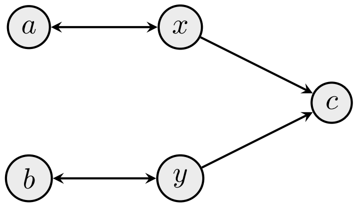

Figure 5.2: AF strictly realizing $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{x\}, \{y\}, \{x, y\}, \{a, b, c\}\}$ under the complete semantics

Theorem 5.12. *The following relations hold:*

$$\Sigma^s_{comp} \not\subseteq \Sigma_{adm},$$

$$\Sigma_{adm} \not\subseteq \Sigma^s_{comp}.$$

Proof. In order to show that $\Sigma^s_{comp} \not\subseteq \Sigma_{adm}$ consider the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{x\}, \{y\}, \{x, y\}, \{a, b, c\}\}$. $\mathcal{S}$ is strictly *comp*-realizable as Figure 5.2 shows, hence $\mathcal{S} \in \Sigma^s_{comp}$. But as $\mathcal{S}$ is not adm-closed, $\mathcal{S} \notin \Sigma_{adm}$, and therefore $\Sigma^s_{comp} \not\subseteq \Sigma_{adm}$.

In order to show that $\Sigma_{adm} \not\subseteq \Sigma^s_{comp}$ consider the extension-set $\mathcal{T} = \{\emptyset, \{a\}, \{a, b\}\}$. We have seen in Proposition 4.20 that $\mathcal{T}$ is adm-realizable and in Proposition 4.24 that $\mathcal{T}$ is not strictly comp-realizable, therefore $\Sigma_{adm} \not\subseteq \Sigma^s_{comp}$. □

Note that for the extension-set $\mathcal{S} = \{\emptyset, \{a\}, \{b\}, \{x\}, \{y\}, \{x, y\}, \{a, b, c\}\}$ we have seen the AF $F^{comp}_{\mathcal{S}}$ obtained from Construction 4.5 in Figure 4.10. $F^{comp}_{\mathcal{S}}$ also realizes $\mathcal{S}$ under the complete semantics. The AF in Figure 5.2 does this in a more compact way. This already suggests that for a certain amount of extension-sets, particularly ones which are strictly realizable, constructions using none or less additional arguments can be found.

Theorem 5.13. *The following relation holds:*

$$\Sigma^s_{comp} \subset \Sigma_{comp}$$

Proof. While $\Sigma^s_{comp} \subseteq \Sigma_{comp}$ trivially holds, we have seen $\Sigma^s_{comp} \not\supseteq \Sigma_{comp}$ in Proposition 4.24. □

So far we have mostly given relations between naive, stage, stable, preferred and semi-stable semantics on the one side and between conflict-free, admissible and complete semantics on the other side. As extension-sets of the former semantics share incomparability, the latter semantics do not. When considering the $\emptyset$-extended signatures (cf. Definition 5.3) we get the following relations.

Theorem 5.14. *The following relations hold:*

$$\Sigma^\emptyset_{naive} \subset \Sigma^\emptyset_{stage} = \Sigma^\emptyset_{stb} \subset \Sigma^\emptyset_{pref} = \Sigma^\emptyset_{sem} \subset \Sigma^\emptyset_{adm} \subset \Sigma^\emptyset_{comp}.$$

Proof. The relations $\Sigma^{\emptyset}_{naive} \subset \Sigma^{\emptyset}_{stage}$, $\Sigma^{\emptyset}_{stage} \subset \Sigma^{\emptyset}_{pref}$ and $\Sigma^{\emptyset}_{pref} = \Sigma^{\emptyset}_{sem}$ follow immediately from Theorem 5.4.

As $\emptyset \notin \Sigma^{\emptyset}_{stb}$ by definition, $\Sigma^{\emptyset}_{stage} = \Sigma^{\emptyset}_{stb}$ follows from Theorem 5.5.

In order to show $\Sigma^{\emptyset}_{pref} \subset \Sigma^{\emptyset}_{adm}$, consider some $\mathcal{S} \in \Sigma^{\emptyset}_{pref}$ and let $\mathcal{S}' = \mathcal{S} \setminus \{\emptyset\}$. $\mathcal{S}'$ is pref-closed and therefore also incomparable. By Lemma 3.9 it holds that $\mathcal{S}' \cup \{\emptyset\} = \mathcal{S}$ is adm-closed. Hence $\mathcal{S} \in \Sigma^{\emptyset}_{adm}$ and therefore $\Sigma^{\emptyset}_{pref} \subset \Sigma^{\emptyset}_{adm}$.

Finally since $\Sigma^{\emptyset}_{adm} = \Sigma_{adm}$, $\Sigma^{\emptyset}_{adm} \subset \Sigma^{\emptyset}_{comp}$ follows directly from Theorem 5.3. □

5.3 Relations by Intertranslatability

In the introductory Chapter 1 we already mentioned the work on intertranslatability of semantics [38, 39, 57] as related to our characterizations of signatures of semantics. Without yet using the term signature, the results on intertranslatability show relations between signatures of semantics. We recall the definition of an exact translation from [39].

Basically a *translation Tr* is a function mapping AFs to AFs. Given two semantics σ and θ, a translation *Tr* is *exact* for $\sigma \Rightarrow \theta$, if for every AF F it holds that $\sigma(F) = \theta(Tr(F))$.

This definition immediately leads to the following proposition.

Proposition 5.15. *Given two semantics σ and θ, it holds that if there exists an exact translation for $\sigma \Rightarrow \theta$, then $\Sigma_\sigma \subseteq \Sigma_\theta$.*

Proof. Consider two semantics σ and θ and a translation *Tr* which is exact for $\sigma \Rightarrow \theta$. Further consider some extension-set $\mathcal{S} \in \Sigma_\sigma$. By Definition 5.1 of signatures there is some AF F with $\sigma(F) = \mathcal{S}$. Now as *Tr* is an exact translation for $\sigma \Rightarrow \theta$, i.e. $\theta(Tr(F)) = \mathcal{S}$, which means that $Tr(F)$ realizes $\mathcal{S}$ under the semantics θ, i.e. $\mathcal{S} \in \Sigma_\theta$, and therefore $\Sigma_\sigma \subseteq \Sigma_\theta$. □

As an example we recall the definition of translation Tr_1 from [39], which is proven to be exact for *pref* $\Rightarrow$ *sem* and *adm* $\Rightarrow$ *comp*. The translation Tr_1 is defined as $Tr_1(F) = (A^*, R^*)$, where

$$A^* = A_F \cup \{a' \mid a \in A_F\}, \text{ and}$$
$$R^* = R_F \cup \{(a,a'),(a',a),(a',a') \mid a \in A_F\}.$$

Figure 5.3 shows Tr_1 on the standard AF F_s from Example 2.2 we used in Chapter 2.

In [39] the authors restrict themselves to computationally efficient translations and give, besides Tr_1 which is exact for *pref* $\Rightarrow$ *sem* and *adm* $\Rightarrow$ *comp*, an exact translation for *stage* $\Rightarrow$ *sem*. Exact translations for *naive* $\Rightarrow$ *stage*, *sem* $\Rightarrow$ *pref*, and *cf* $\Rightarrow$ *adm* are given in [38], where also possibly inefficient translations are taken into account.

Putting together these results already gives the following relations between signatures of semantics:

$$\Sigma_{naive} \subseteq \Sigma_{stage} \subseteq \Sigma_{pref} = \Sigma_{sem}$$

and

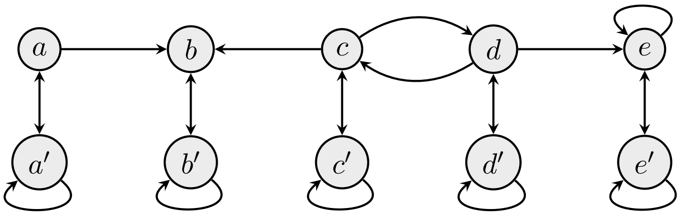

Figure 5.3: $Tr_1(F_s)$ for the AF F_s from Example 2.2

$$\Sigma_{cf} \subseteq \Sigma_{adm} \subseteq \Sigma_{comp}.$$

The characterizations of signatures given in this thesis add to the insights gained from inter-translatability results in the following ways:

- First of all, the $\subseteq$-relations are proven to be $\subset$-relations by giving examples that the other side of the set-inclusion cannot hold. Theorem 5.4 summarizes these results.
- Second and most important, the results on signatures of semantics not only show the relations between signatures, but give exact characterizations of the signatures, i.e. describe, which extension-sets have an AF as counterpart under a certain semantics. This makes the results beneficial for applications such as model-based revision or pruning of search-space.
- By characterizing the signatures of semantics the relations between signatures, which were known because of the existence of translations, are now shown to hold because of the properties each extension-set in the signature of a certain semantics fulfills. This leads to a better understanding of why the relations hold and what the difference between signatures of two semantics is.

5.4 Discussion

The Venn diagram in Figure 5.4 depicts the relations between the signatures of the semantics we considered in this thesis. The set $\Sigma_{\mathcal{A}}$ contains all extension-sets $\mathcal{S} \subseteq 2^{\mathcal{A}}$ where $Args_{\mathcal{S}}$ is finite. The diagram shows the division of extension-sets into those which are in the signatures of conflict-free, admissible, and complete semantics on the one side and these extension-sets which are in the signatures of naive, stable, stage, preferred, and semi-stable semantics on the other side. The dividing criterion is incomparability. Extension-sets of the latter semantics are always incomparable, while ones of the former semantics are not. Moreover an extension-set $\mathcal{S}$ under the conflict-free, admissible and complete semantics is only incomparable if it contains solely the empty extension, i.e. $\mathcal{S} = \{\emptyset\}$. Therefore the only extension-set all signatures share is $\{\emptyset\}$. The signature that steps out of line is the one of the stable semantics. Σ_{stb} is the union of Σ_{stage} and the empty set of extensions.

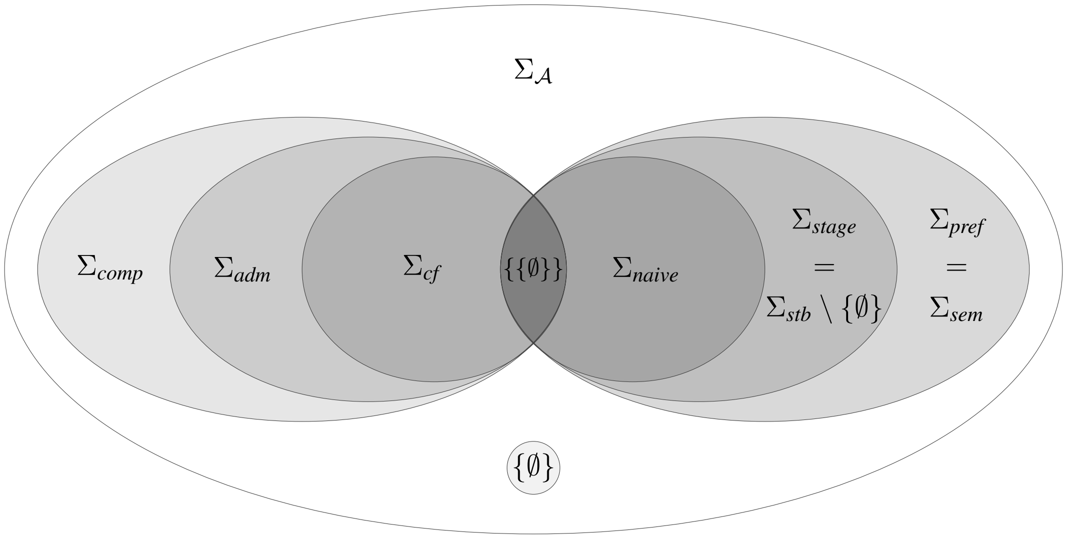

Figure 5.4: Venn-Diagram showing the relations between the expressiveness of semantics

The division criterion of incomparability is left aside when considering the $\emptyset$-extended signatures. The Venn diagram in Figure 5.5 depicts the relations between the $\emptyset$-extended signatures. As each $\mathcal{S} \in \Sigma_{cf}^{\emptyset}$ is downward-closed, $\{\emptyset\}$ is still the only extension-set shared by $\Sigma_{cf}^{\emptyset}$ and $\Sigma_{\sigma}^{\emptyset}$, where σ is an incomparable semantics. The $\emptyset$-extended signatures of admissible and complete semantics are supersets of the signatures of incomparability-based semantics (cf. Theorem 5.14). $\Sigma_{\mathcal{A}}^{\emptyset}$ is the set of all extension-sets $\mathcal{S} = (\mathcal{S}' \cup \{\emptyset\})$ where $\mathcal{S}' \subseteq 2^{\mathcal{A}}$ and $Args_{\mathcal{S}}$ is finite.

Although we have not given exact characterizations for strict signatures except for conflict-free

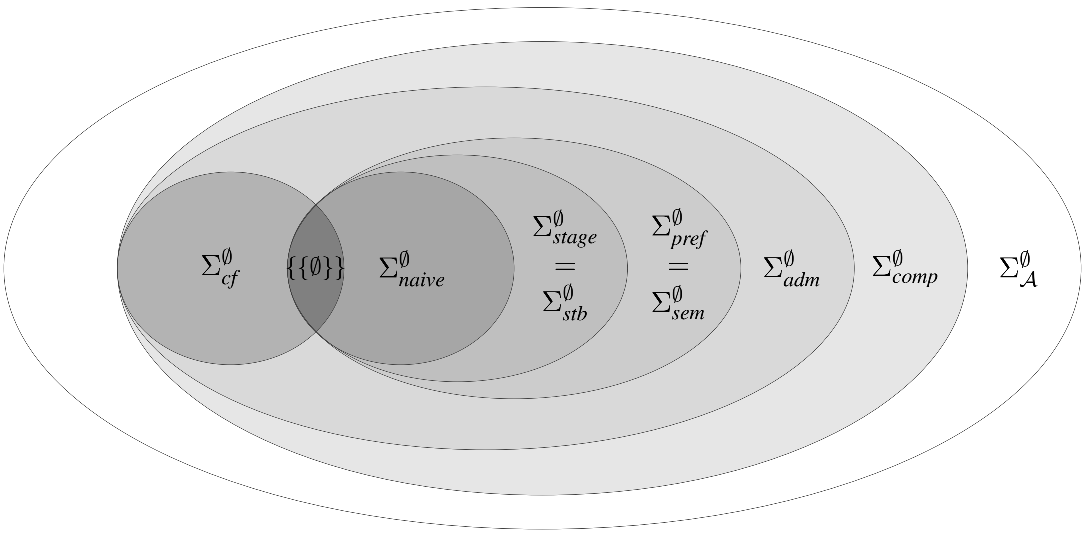

Figure 5.5: Venn-Diagram showing the relation between the $\emptyset$-extended signatures

sets and naive semantics, the results in Section 5.2 already indicate some relations to general signatures. Interestingly, the strict signature of the stage (resp. admissible) semantics is a proper superset of the signature of the naive (resp. conflict-free) semantics (cf. Theorems 5.9 and 5.8), while the strict signature of complete semantics is incomparable to the general signature of the admissible semantics (cf. Theorem 5.12).

Interesting open questions are the following relations between (strict) signatures:

$$\Sigma_{stage} \sim \Sigma^s_{pref}$$
$$\Sigma_{stage} \sim \Sigma^s_{sem}$$
$$\Sigma^s_{sem} \sim \Sigma^s_{pref}$$

By Theorem 5.11 we know that $\Sigma_{stage} \not\subseteq \Sigma^s_{pref}$ and $\Sigma_{stage} \not\subseteq \Sigma^s_{sem}$, but by the lack of certain knowledge we have to be content with being confident that future work will show that $\Sigma_{stage} \supset \Sigma^s_{pref}$ and $\Sigma_{stage} \supset \Sigma^s_{sem}$ hold, i.e. each extension-set $\mathcal{S}$ which is strictly realizable under the preferred and semi-stable semantics is non-empty, incomparable, and tight.

CHAPTER 6

Conclusion

In this final chapter we are going to give a summary of the results achieved in this thesis. We will further reveal the implications we can draw from these results, and finally give an outlook to possible future research directions.

6.1 Summary

To begin with the main contribution, we characterized the signatures Σ_σ for the semantics $\sigma \in \{\mathit{cf}, \mathit{adm}, \mathit{naive}, \mathit{stage}, \mathit{stb}, \mathit{pref}, \mathit{sem}, \mathit{comp}\}$ (cf. Theorem 5.1) and therefore described the expressiveness in terms of multiple viewpoints of each semantics. Knowing exactly about the characterizations of a signature Σ_σ implies the following:

- Each set of extensions $\sigma(F)$ obtained from applying semantics σ to an arbitrary AF F is contained in Σ_σ, i.e. $\sigma(F) \in \Sigma_\sigma$.
- For each $\mathcal{S} \in \Sigma_\sigma$ there is some AF F which realizes $\mathcal{S}$ under σ, i.e. $\sigma(F) = \mathcal{S}$.

Constructions 4.1, 4.2, 4.3, 4.4, and 4.5 give definitions of how to build such AFs realizing a given extension-set under the various semantics.

The signatures are characterized by using the properties defined in Chapter 3 which makes it rather straight-forward to check whether an extension-set is in the signature and can therefore be realized.

The relations between signatures proposed in Section 5.2 gave some insights on the different capabilities of semantics. To summarize some notable results:

- Stable and stage semantics are more expressive than naive semantics.
- The preferred semantics is more expressive than stage and stable semantics[1].

[1] With the exception of $\emptyset$, which is only realizable by the stable semantics.

- The stable semantics is as expressive as the stage semantics[2].
- The preferred semantics is as expressive as the semi-stable semantics.
- The complete semantics is more expressive than the admissible semantics.

We further strengthened the notion of a signature by introducing the strict signature Σ^s_σ which is the collection of extension-sets where one can find an AF F which is solely built from arguments occurring in $\mathcal{S}$ and delivers $\sigma(F) = \mathcal{S}$. In other words no additional arguments to express $\mathcal{S}$ are required. We showed that $\Sigma^s_\sigma = \Sigma_\sigma$ holds for $\sigma \in \{\mathit{cf}, \mathit{naive}\}$, while for all other semantics of our interest we showed that $\Sigma^s_\sigma \subsetneq \Sigma_\sigma$. Moreover, we showed that e.g. not even $\Sigma_{\mathit{stage}} \subseteq \Sigma^s_{\mathit{pref}}$ holds, i.e. there exists an extension-set which is realizable under the stage semantics, but not strictly realizable under the preferred semantics.

6.2 Implications

In Section 1.2 we pointed out the importance of signatures when it comes to model-based belief revision. Knowing whether a set of extensions $\mathcal{S}$ is contained in Σ_σ is a necessary condition which should be checked before actually looking for an AF F which realizes $\mathcal{S}$ under σ. In an example in the introductory Chapter 1 we showed that some revised extension-set was not realizable with the preferred semantics, since preferred extensions are incomparable. We have seen in this thesis that the expressive power of semantics differs from semantics to semantics not only by incomparability. The following example makes this more concrete.

Example 6.1. Consider the framework F in Figure 6.1. First note that $\mathit{stb}(F) = \{\{a, c, e\}, \{b, d, e\}\}$. Now one asks for adaption of the framework such that the extension is revised in a way that also $\{a, b\}$ is a stable extension, i.e. the desired set of extensions $\mathcal{S}_4 = \{\{a, b\}, \{a, c, e\}, \{b, d, e\}\}$ (see Example 3.1). This is not possible as $\mathcal{S}_4$ is not tight (cf. Example 3.3) and therefore not realizable under the stable semantics.

However, this case behaves different with the preferred semantics. First note that $\mathit{pref}(F) = \mathit{stb}(F)$ holds for the AF F in Figure 6.1. Moreover with preferred semantics one can meet the desires of adaption of F such that $\{a, b\}$ is an extension too. The AF F_p in Figure 3.3 has $\mathcal{S}_4$ as its preferred extensions. One can verify that the desired set of extensions $\mathcal{S}_4$ is realizable under the preferred semantics as $\mathcal{S}_4$ is pref-closed (see Example 3.6).

In the introductory chapter we also mentioned that being aware of the signature of a semantics, a concrete implementation can prune its search-space according to the characterization of the signature. In an example in Chapter 1 we showed this for a case where all sub- and supersets of an already found extension could be removed from the set of possible extension-candidates because of incomparability.

Having the exact characteristics of signatures at hand, one can do more involved pruning of search space. The following example shows how pruning can be done when computing the complete extensions, knowing that the set of complete extensions is comp-closed (cf. Definition 3.10).

[2] With the exception of $\emptyset$, which is only realizable by the stable semantics.

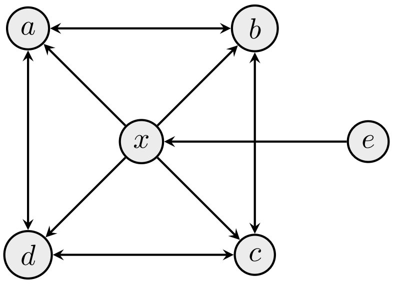

Figure 6.1: AF F with $stb(F) = pref(F) = \{\{a,c,e\},\{b,d,e\}\}$

Example 6.2. Consider an AF $F = (A, R)$ and a procedure P which determines all complete extensions of F, i.e. $comp(F)$. Now assume P has found $S_1 = \{a\}$, $S_2 = \{b\}$, and $S_3 = \{a,b,c\}$ as complete extensions and revealed that $\{a,b\}$ is no complete extension. As $comp(F)$ is comp-closed by Proposition 3.20 there exists a unique completion-set $C_{comp(F)}(\{a\} \cup \{b\}) \subseteq \{a,b\}$. Since $\{a,b\} \notin comp(F)$ it follows that $C_{comp(F)}(\{a\} \cup \{b\}) = \{a,b,c\}$. Therefore the search-space of P can be pruned in a way that all S', where $S' \supset \{a,b\}$ and $c \notin S'$, are removed from the set of extension-candidates.

6.3 Future Work

Future work on the topic of realizability and signatures could go in various directions.

First of all the exact characterization of the strict signature of stable, stage, preferred, semi-stable, and complete semantics is still open. Strict signatures are important in cases where the application does not allow self-attacking arguments or no additional arguments at all.

Secondly, we have only taken into account some argumentation semantics, the most heavily studied though. Future work could investigate the signatures of other semantics, such as the *cf2*-semantics [8, 45], and the *resolution-based grounded* semantics [4]. Especially the results on resolution-based grounded semantics appear to be interesting, as their extensions meet all the properties defined in [5] and therefore the semantics is seen as the "best" in that context. A characterization of the signature of the resolution-based grounded semantics would shed a light on to which extent the quality in the sense of [5] influences the expressiveness of the semantics.

Since we have viewed semantics here only in an extension-based way, it would also be of high interest to extend our studies to labelling-based semantics [25]. While with extension-based semantics, an extension defines itself by the arguments which are in the extension and these which are not, the application of a semantics in the labelling-based way of [25] provides a set of labellings, where each labelling maps one of three labels to each argument: the label "in" to the arguments in the corresponding extension; the label "out" to those being attacked by "in"-arguments; and the label "undec" to all other arguments. In this context it is e.g. unlikely that the signatures of stable and stage semantics coincide as they nearly[3] do in the extension-based context.

[3]Except $\emptyset$.

Moreover there are several extensions to Dung's argumentation frameworks, where it is worth investigating on the expressiveness of semantics. Examples, to name some of them, are value-based argumentation frameworks [11], bipolar argumentation frameworks [28] and abstract dialectical frameworks [18, 19].

Another important open issue is complexity analysis. Especially, as one particular application of our results is the problem of recasting, i.e. to decide whether the σ-extensions of a given AF can be expressed via a different semantics θ. The relevance of this problem is, for instance, given by the fact that θ is a semantics for which we have faster systems available. Complexity results of the recasting problem are of high interest, they most likely go up to Π_2^P-completeness at least.

The provided definitions of canonical argumentation frameworks could give rise to finding appropriate normal-forms for AFs. The fact that, given a certain semantics, each extension-set in the corresponding signature can be realized by an AF obtained by using a uniform construction, makes it clear, that a very limited amount of AFs suffices in order to capture all realizable extension-sets. A definition of normal-forms together with proper translations could be beneficial. Similar work in the field of logic programming has been done in [17].

Finally, current implementations of argumentation semantics can use the result on signatures in order to prune search-space as described in Section 1.2 and made concrete in Example 6.2.

Bibliography

[1] Katie Atkinson. Value-based argumentation for democratic decision support. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Computational Models of Argument: Proceedings of COMMA 2006*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 47–58. IOS Press, 2006.

[2] Katie Atkinson, Trevor J. M. Bench-Capon, and Peter McBurney. Parmenides: Facilitating deliberation in democracies. *Artificial Intelligence and Law*, 14(4):261–275, 2006.

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

[4] Pietro Baroni, Paul E. Dunne, and Massimiliano Giacomin. On the resolution-based family of abstract argumentation semantics and its grounded instance. *Artificial Intelligence*, 175(3-4):791–813, 2011.

[5] Pietro Baroni and Massimiliano Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.

[6] Pietro Baroni and Massimiliano Giacomin. A systematic classification of argumentation frameworks where semantics agree. In Philippe Besnard, Sylvie Doutre, and Anthony Hunter, editors, *Computational Models of Argument: Proceedings of COMMA 2008*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 37–48. IOS Press, 2008.

[7] Pietro Baroni and Massimiliano Giacomin. Semantics of abstract argument systems. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 25–44. Springer US, 2009.

[8] Pietro Baroni, Massimiliano Giacomin, and Giovanni Guida. SCC-Recursiveness: A general schema for argumentation semantics. *Artificial Intelligence*, 168(1-2):162–210, 2005.

[9] Ringo Baumann. What does it take to enforce an argument? minimal change in abstract argumentation. In Luc De Raedt, Christian Bessière, Didier Dubois, Patrick Doherty, Paolo Frasconi, Fredrik Heintz, and Peter J. F. Lucas, editors, *ECAI 2012 - 20th European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 127–132. IOS Press, 2012.

[10] Ringo Baumann and Gerhard Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Computational Models of Argument: Proceedings of COMMA 2010*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 75–86. IOS Press, 2010.

[11] Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

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

[13] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In James P. Delgrande and Torsten Schaub, editors, *10th International Workshop on Non-Monotonic Reasoning (NMR 2004), Proceedings*, pages 59–64, 2004.

[14] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. MIT Press, 2008.

[15] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Duality between addition and removal - a tool for studying change in argumentation. In Salvatore Greco, Bernadette Bouchon-Meunier, Giulianella Coletti, Mario Fedrizzi, Benedetto Matarazzo, and Ronald R. Yager, editors, *Advances on Computational Intelligence - 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2012, Proceedings, Part I*, volume 297 of *Communications in Computer and Information Science*, pages 219–229. Springer, 2012.

[16] Richard Booth, Martin Caminada, Mikolaj Podlaszewski, and Iyad Rahwan. Quantifying disagreement in argument-based reasoning. In Wiebe van der Hoek, Lin Padgham, Vincent Conitzer, and Michael Winikoff, editors, *International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2012, Proceedings*, pages 493–500. IFAAMAS, 2012.

[17] Stefan Brass and Jürgen Dix. Semantics of (disjunctive) logic programs based on partial evaluation. *Journal of Logic Programming*, 40(1):1–46, 1999.

[18] Gerhard Brewka, Paul E. Dunne, and Stefan Woltran. Relating the semantics of abstract dialectical frameworks and standard AFs. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 780–785. IJCAI/AAAI, 2011.

[19] 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*. AAAI Press, 2010.

[20] Hans Kleine Büning and Theodor Lettmann. *Aussagenlogik - Deduktion und Algorithmen*. Leitfäden und Monographien der Informatik. Teubner, 1994.

[21] Martin Caminada. Semi-stable semantics. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Computational Models of Argument: Proceedings of COMMA 2006*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 121–130. IOS Press, 2006.

[22] Martin Caminada. Comparing two unique extension semantics for formal argumentation: ideal and eager. In *Proceedings of the 19th Belgian-Dutch Conference on Artificial Intelligence (BNAIC 2007)*, pages 81–87, 2007.

[23] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.

[24] Martin Caminada, Walter Alexandre Carnielli, and Paul E. Dunne. Semi-stable semantics. *Journal of Logic and Computation*, 22(5):1207–1254, 2012.

[25] Martin Caminada and Dov M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2):109–145, 2009.

[26] Martin Caminada and Bart Verheij. On the existence of semi-stable extensions. *Proceedings of the 22nd Benelux Conference on Artificial Intelligence (BNAIC 2010)*, 2010.

[27] Dan Cartwright and Katie Atkinson. Political engagement through tools for argumentation. In Philippe Besnard, Sylvie Doutre, and Anthony Hunter, editors, *Computational Models of Argument: Proceedings of COMMA 2008*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 116–127. IOS Press, 2008.

[28] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolar abstract argumentation systems. In *Argumentation in Artificial Intelligence*, pages 65–84. Springer, 2009.

[29] Sylvie Coste-Marquis, Caroline Devred, and Pierre Marquis. Symmetric argumentation frameworks. In Lluis Godo, editor, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 8th European Conference, Proceedings*, volume 3571 of *Lecture Notes in Computer Science*, pages 317–328. Springer, 2005.

[30] Nadia Creignou, Odile Papini, Reinhard Pichler, and Stefan Woltran. Belief revision within fragments of propositional logic. In Gerhard Brewka, Thomas Eiter, and Sheila A. McIlraith, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference*. AAAI Press, 2012.

[31] James P. Delgrande and Pavlos Peppas. Revising horn theories. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 839–844. IJCAI/AAAI, 2011.

[32] Sylvie Doutre and Jérôme Mengin. Preferred extensions of argumentation frameworks: Query answering and computation. In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *Automated Reasoning, First International Joint Conference, IJCAR 2001, Proceedings*, volume 2083 of *Lecture Notes in Computer Science*, pages 272–288. Springer, 2001.

[33] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[34] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10-15):642–674, 2007.

[35] Paul E. Dunne and Michael Wooldridge. Complexity of abstract argumentation. In *Argumentation in Artificial Intelligence*, pages 85–104. Springer, 2009.

[36] Wolfgang Dvořák. *Computational Aspects of Abstract Argumentation*. PhD thesis, Vienna University of Technology, 2012.

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

[38] Wolfgang Dvořák and Christof Spanring. Comparing the expressiveness of argumentation semantics. In Bart Verheij, Stefan Szeider, and Stefan Woltran, editors, *Computational Models of Argument - Proceedings of COMMA 2012*, volume 245 of *Frontiers in Artificial Intelligence and Applications*, pages 261–272. IOS Press, 2012.

[39] Wolfgang Dvořák and Stefan Woltran. On the intertranslatability of argumentation semantics. *Journal of Artificial Intelligence Research (JAIR)*, 41:445–475, 2011.

[40] Uwe Egly, Sarah Alice Gaggl, and Stefan Woltran. Answer-set programming encodings for argumentation frameworks. *Argument and Computation*, 1(2):147–177, 2010.

[41] Uwe Egly and Stefan Woltran. Reasoning in argumentation frameworks using quantified boolean formulas. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Computational Models of Argument: Proceedings of COMMA 2006*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 133–144. IOS Press, 2006.

[42] Thomas Eiter, Michael Fink, and Stefan Woltran. Semantical characterizations and complexity of equivalences in answer set programming. *ACM Transactions on Computational Logic (TOCL)*, 8(3), 2007.

[43] Herbert B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 1972.

[44] Marcelo A. Falappa, Alejandro Javier García, Gabriele Kern-Isberner, and Guillermo Ricardo Simari. On the evolving relation between belief revision and argumentation. *Knowledge Engineering Review*, 26(1):35–43, 2011.

[45] Sarah Alice Gaggl and Stefan Woltran. cf2 semantics revisited. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Computational Models of Argument: Proceedings of COMMA 2010*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 243–254. IOS Press, 2010.

[46] Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *Proceedings of the 5th International Conference on Logic programming*, pages 1070–1080, 1988.

[47] Hadassa Jakobovits and Dirk Vermeir. Dialectic semantics for argumentation frameworks. In *Proceedings of the 7th International Conference on Artificial intelligence and Law*, pages 53–62. ACM, 1999.

[48] Diego C. Martínez, Alejandro Javier García, and Guillermo Ricardo Simari. On acceptability in abstract argumentation frameworks with an extended defeat relation. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Computational Models of Argument: Proceedings of COMMA 2006*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 273–278. IOS Press, 2006.

[49] Peter Bro Miltersen, Jaikumar Radhakrishnan, and Ingo Wegener. On converting CNF to DNF. In Branislav Rovan and Peter Vojtás, editors, *Mathematical Foundations of Computer Science 2003, 28th International Symposium, Proceedings*, pages 612–621, 2003.

[50] Sanjay Modgil and Martin Caminada. Proof theories and algorithms for abstract argumentation frameworks. In *Argumentation in Artificial Intelligence*, pages 105–129. Springer, 2009.

[51] Sanjay Modgil and Henry Prakken. Revisiting preferences and argumentation. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1021–1026. IJCAI/AAAI, 2011.

[52] Robert C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25(1):75–94, 1985.

[53] Henry Prakken. A logical framework for modelling legal argument. In *Proceedings of the 4th International Conference on Artificial Intelligence and Law*, pages 1–9. ACM, 1993.

[54] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.

[55] Iyad Rahwan and Guillermo R. Simari, editors. *Argumentation in Artificial Intelligence*. Springer, 2009.

[56] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1–2):81–132, 1980.

[57] Christof Spanring. Intertranslatability Results for Abstract Argumentation Semantics. Master's thesis, Vienna University of Technology, 2012.

[58] Katia P. Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28(3):203–242, 1990.

[59] Francesca Toni and Marek Sergot. Argumentation and answer set programming. In *Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning*, pages 164–180. Springer, 2011.

[60] Bart Verheij. Two approaches to dialectical argumentation: admissible sets and argumentation stages. In John-Jules C. Meyer and Linda C. van der Gaag, editors, *Proceedings of the 8th Dutch Conference on Artificial Intelligence*, pages 357–368, 1996.

[61] Bart Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation*, 13(3):319–346, 2003.

[62] Bruno Zanuttini and Jean-Jacques Hébrard. A unified framework for structure identification. *Information Processing Letters*, 81(6):335–339, 2002.