Argumentation, Logic and Explainability

Théo Duchatelle

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Théo DUCHATELLE

Le 13 décembre 2023

Argumentation, Logique et Explicabilité

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Informatique et Télécommunications**

Unité de recherche :
IRIT : Institut de Recherche en Informatique de Toulouse

Thèse dirigée par
Marie-christine LAGASQUIE

Jury

M. Nicolas MAUDET, Rapporteur
M. Stefan WOLTRAN, Rapporteur
Mme Sylvie DOUTRE, Examinatrice
M. Bruno ZANUTTINI, Examinateur
Mme Leila AMGOUD, Examinatrice
Mme Marie-christine LAGASQUIE, Directrice de thèse

Argumentation, Logique et Explicabilité

Explications pour l'Argumentation Abstraite : du Visuel à la Logique

L'Argumentation Abstraite en tant que moyen de prise de décision est un domaine de l'informatique recevant de plus en plus d'attention. Formellement, on considère un ensemble d'objets abstraits appelés "arguments" et une relation de conflit entre eux. La prise de décision s'effectue en identifiant des groupes d'arguments ayant de bonnes propriétés vis-à-vis de la relation de conflit. Cependant, les techniques employées ne produisent pas de résultats pouvant être justifiés par des explications intuitives pour des utilisateurs humains. Les travaux menés dans cette thèse visent à produire des explications pour ces processus de décision qui soient accessibles aux non experts, en particulier en tâchant de se reposer sur des critères visuels. De plus, les travaux menés ambitionnent de généraliser les résultats obtenus dans des contextes d'argumentation enrichis. En effet, il est possible d'étendre le cadre de base en ajoutant une relation supplémentaire positive (de support) entre les arguments, de considérer des relations d'ordre supérieur ou de considérer des arguments agissant en coalitions. Chaque enrichissement change la façon de questionner et d'accepter les arguments. Pour finir, une implémentation des concepts proposés sera fournie via l'encodage logique des cadres d'argumentation, de leurs processus de décision et de leurs explications.

Argumentation, Logic and Explainability

Explanations for Abstract Argumentation: from Visual to Logic

Abstract Argumentation, as a decision tool, currently is a research topic undergoing intense study. Formally, we consider a set of abstract objects that we call "arguments" and a conflict relation between them. The decision is made by identifying sets of arguments that have desirable properties regarding the conflict relation. However, systems based on these formal models provide outputs that can hardly be supported by explanations perceived as intuitive by human users. The work carried out aim at providing explanations for these decision processes that are understandable even by non experts, in particular by trying to rely on visual criteria. What's more, we also aim at generalising the results we obtain to the context of enriched argumentation frameworks. Indeed, it is possible to extend the basic framework by adding an additional positive relation (a support relation) between the arguments, by considering higher order relations, or by considering arguments that work in coalitions. Each of these enrichments changes the way the acceptability and non-acceptability of arguments is questioned. To complete the picture, an implementation of the proposed concepts will be made through the logical encoding of the argumentation frameworks, their decision processes and their explanations.

Acknowledgements

First and foremost, I would like to express my gratitude towards my supervisors, Marie-Christine Lagasquie-Schiex and Sylvie Doutre. Not only have they provided me with invaluable insights and advice for my research, but they also always made sure that it was ultimately me who decided the direction my works would take. Having had discussions with other PhD. students made me realise how scarce this is, so I cannot be thankful enough to them for ensuring the best conditions I could have hoped for during my three years of work. I would also like to express my gratitude to Philippe Besnard, who supervised me during the beginning of my PhD but had to retire before its end, for his wise guidance into the world of academic research.

I would then like to thank my reviewers, Nicolas Maudet and Stefan Woltran, for their work and investment in reviewing my dissertation. Their remarks and suggestions have unveiled some aspects of my work that I did not suspect and promise exciting research directions for the future. Likewise, I want to thank my examiners, Bruno Zanuttini and Leila Amgoud, for the interest they showed in my works, for their presence at my defence, and for the enlightening discussions we had after my presentation.

Finally, I am especially grateful to my family, my mother, my two brothers and our three cats, for their unconditional and continuous support and love throughout my PhD study. Without their understanding and encouragement during this period, I would probably not have been able to achieve my work. My gratitude and appreciation extends to all my close friends that I hold dear who, be it around a drink, a game, both, or any other occasion found a way to make me keep going always onward.

Contents

Chapter 1

Introduction

When it comes to taking decisions using Artificial Intelligence (AI) techniques, an obvious observation is the ever increasing complexity of these techniques. The most telling example is probably the recent rise and general deployment of artificial neural networks techniques across an extremely wide spectrum of applications. Although the first perceptron, that is to say the first artificial model of a biological neuron, was theorized as far back as in the 1950s ([Ros58]), such methods could not possibly be used until recently, due to a lack of computational power. Now, however, this computational power is available, and with it neural networks have achieved results that were previously thought to be unreachable. This of course resulted in neural networks (and AI in general, but mostly neural networks) being largely spread and discussed among a way larger audience than the restricted club of field specialists: non-expert people. So much so actually, that more usually than not for non-experts, the term "AI" solely refers to neural networks. It is highly probable that a random non-expert does not know, for instance, what the $A*$ algorithm is or how it works, and even if it is described, it will probably (wrongly) not be considered as "AI".

Nonetheless, all this generalised interest and curiosity about neural networks have unveiled a troublesome and somewhat worrying observation. Indeed, neural networks were very promising, extremely efficient, and applicable on a vast variety of areas. Thus, people wanted to know the inner workings of neural networks so that they could effectively be applied. So they inquired the experts to describe to them what was going on when a neural network algorithm was running. And there came the troublesome observation: *even the experts did not know what was going on.* Now, that is of course a bit of an exaggeration. Experts do know the general principles that guide the execution and use of a neural network and there are situations in which a description of the network's behavior can be satisfying enough. But, more usually than not, once a neural network has been trained and is being used, experts are unable to tell precisely what makes it take a decision or another. This is an important change of paradigm because, before neural networks, obtaining such a description and understanding of the inner workings of an algorithm was always possible. With neural networks however, this is not always possible.

The importance of this flaw should not be underestimated. Among other consequences, this means in particular that the safety of a neural networks cannot be assessed. In other terms, there are no ways to be sure that a neural networks does what it is required to do, and more importantly, if it fails to do so, there are no ways to know precisely where such a failure comes from. Consequently, it is usually considered unacceptable to let a neural network assume a critical function, on which could depend human lives for instance. Furthermore, such a lack of reliability has in fact tempered the willingness of (generally private) actors to even use neural networks, be it for critical or safe functions. As a consequence, the general deployment of neural networks is going at a slower pace than one could have initially imagined.

And yet, such a general deployment is still desirable to and desired by many who put a great deal of efforts in that direction. And, as would be expected now, one of these efforts is to find ways to know what is going on inside neural networks. To put it simply, a lot of researches that wish to make neural networks more understandable are looking for ways to define *explanations*. As a matter of fact, this quest for explanations

for neural networks has given rise to its own dedicated area of research: Explainable Artificial Intelligence (XAI). To measure the importance of XAI, we point out two indicators: (1) XAI has been the subject of a research project by the DARPA back in 2016 ([DAR16]), and (2) AI has been the subject of a regulation in the European Union (EU), first in EU's General Data Protection Regulation (GDPR) in 2018 ([Uni16]), and then in a proposition by the European Commission (the AI Act, [Com21]) in 2021 and still subject to debates. One of the objectives of both EU's regulations is for individuals to have a "right to explanations" when dealing with AI systems.

The intense research being undertaken on the problem of finding good explanations for neural networks have already yielded a lot of results, among which are explainability methods. The most popular ones are probably LIME ([RSG16]) and SHAP ([LL17]). To present them in a few words, LIME locally approximates the decision taken by the initial decision-making process by using a simpler and understandable decision-making process, while SHAP instead computes a score for each input that represents the impact of that input on a precise decision. Both these methods have a certain number of qualities. To give only one, they are both model-agnostic, which they can be used not only for neural networks but in fact for any machine learning algorithm, of which neural networks are only one representative. However, there are also important issues with both methods. In the case of LIME, since it produces a local approximation of the initial decision-making process, it is not faithful to the overall algorithm. Sure, it might give targeted insights at how the algorithm behaves on specific inputs, but it still provides no insight as to how it works in general. In the case of SHAP, it could be said to not give enough details. What good is knowing which inputs played the more important roles if we do not know *which* role was played and *how*? As a matter of fact, these problems are not exclusive only to LIME and SHAP, but to many explanation methods, as pointed in [Rud19]. Based on these observations (among others), the author of [Rud19] advocates for not using algorithms that would require explanations to become interpretable, and to use algorithms that are directly interpretable instead (for critical functions at least). But is being interpretable enough? The author of [Rud19] notes that interpretability is a domain-specific notion. Yet, it somehow generally refers to the ability of peeking inside an algorithm and see what it is made of. Is knowing what constitutes an algorithm enough to make it understandable? We consider that it is not the case. To motivate our answer, consider expert systems. They are interpretable, it is possible to say what they are made of: facts, rules, and an inference system. Moreover, most inference systems are in fact relying on deductions based on a formal logic representation of knowledge. And still, despite being interpretable, researchers on the area sought to give them explanatory capabilities, way before such capabilities were also sought for neural networks (see for instance the survey of [MS88] dating from the 1980s). This is not in contradiction with the observation that the quest for explanations for neural networks gave rise to XAI. With neural networks came an explosion of interest on this research problematic. However, that problematic existed and was studied before this explosion of attention, even in computer science. Also, on a side note, it is remarkable to see how the reasons that pushed researchers to have explanations for expert systems are *similar* to the reasons that now push researchers to have explanations for neural networks, even though the two systems are *radically different*. It eventually comes down to: "If we want this system (i.e. expert systems / neural networks) to be more widely used, we must increase the trust people have in it by making it more understandable". To us, this is related to the misconception that interpretability is enough to make a computer program understandable by potentially anyone. To us, interpretability is instead enough to make a computer program understandable by *experts*. This is why we still consider relevant the ability to produce explanations, even for interpretable systems. Because, anyone who might come to use it, and who just happens to not have studied computer science that much, will still lack the *expert knowledge* required to understand what is really going on, thus resulting in distrust in the system. For that kind of people, explanations are relevant and useful, no matter what kind of system is being used. We also believe that this is why there was such a burst of interest when this problematic emerged for neural networks: in this case, even the experts cannot understand what is going on.

Here is thus the purpose we believe explanations should serve: to breach the gap created by expert knowledge and to make accessible what would otherwise be exclusively reserved to experts. We do believe that neural networks (and other non interpretable machine learning techniques) are the gateway to all sorts

of new exciting possibilities in the domain of AI. However, the exploration of these possibilities and the deployment of applications emerging from them direly needs for potentially anyone to know what is being done exactly (even if explorations and applications are certainly being made without this requirement being met). Thus, in our opinion, research on the matter should follow two tracks, parallel at first, but that will eventually meet: firstly, to make non interpretable machine learning interpretable (so understandable for experts) and secondly to make interpretable systems explainable (so understandable for non experts).

Now, after our rather expansive contextualization and exposition of beliefs, let us focus on the present work. It should be seen as a part of the second track of research we outlined previously. As such, we do not work on neural networks or any other non interpretable machine learning formalism. Instead, the subject of our research is the definition of explanations for systems that are already interpretable. The hope we have, and that we left implicit at the end of the previous paragraph, is that by working towards making interpretable systems explainable one by one, a general theory of explanation for interpretable formalisms could emerge. Such a theory would be of great use, say when non interpretable machine learning techniques would become interpretable for instance. So, we are actually interested here in producing explanations for a specific interpretable formalism: *Abstract Argumentation*.

The formalism known as Abstract Argumentation is being increasingly studied to provide explanations in all sorts of domains. As a testimony of this trend, the survey [ČRA+21] cites works whose objective is to build explanations, using Abstract Argumentation, for systems in fields such as Classification, Recommender systems or Planning and Scheduling. As the authors of the survey note, in these cases, Abstract Argumentation is chosen to provide explanations because it intrinsically possesses some properties that are perceived as desirable when explaining. Indeed, Argumentation Frameworks (the main objects handled in Abstract Argumentation) natively have dialectical aspects: it is very easy, by their nature, to project a discussion on them. This makes Abstract Argumentation coherent with the observation from social sciences that explanations are social, that they are transferred in a conversation by someone to someone else. Additionally, decisions taken using Abstract Argumentation are made so that they are "robust" in some sense. The criteria used to make these decisions offer a high capability of convincing that the decision is coherent, justifiable, dominant, ... depending on the criteria used.

Abstract Argumentation is what we call a non-monotonic reasoning formalism. In mathematics, monotony is a concept that refers to a certain idea of stability. When we say that something is monotonic, it usually means that it stays the way it is, at least regarding a particular property. As such, non-monotony would in turn refer to a certain idea of instability. Now, a reasoning formalism is a mathematical tool whose aim is to model and mimic human reason. Thus, reasoning formalisms are usually designed to allow, one way or another, to infer new knowledge from a given base of knowledge (which is incidentally usually called a "Knowledge Base"). As such, in a non-monotonic reasoning formalism, like Abstract Argumentation, when we make further observations after having inferred some conclusions, these conclusions may not be valid anymore: they are, in a sense, unstable.

The usual take in a non-monotonic reasoning formalism is to consider that our Knowledge Base contains conflicting observations from the start (we say that it is inconsistent). There are then several ways to deal with this situation and try to draw conclusions despite our starting inconsistency. The earliest works of the domain focused on trying to find the maximum amount of non-conflicting information in the Knowledge Base. We could then proceed with a monotonic reasoning formalism from there on. Other works later focused on trying to repair the inconsistent Knowledge Base, and try to make it consistent again. To do so, these works actually aimed at finding the least amount of information that was conflicting, so that it could get rid off in one way or another to get closer to a consistent Knowledge Base.

The study of argumentation from a computer science perspective originates from the latter family of works in non-monotonic reasoning formalisms. However, despite focusing on finding the minimal conflicts in a Knowledge Base, the primary objective of argumentation was not necessarily to repair that Knowledge Base. In fact, the study of argumentation in computer science rapidly followed two parallel, but not uncorrelated, tracks. The first one, Structured Argumentation, is fully integrated in the family of works that aim at finding minimal amounts of conflicting information. More precisely, from the inconsistent Knowledge Base, Structured Argumentation has ways to produce what is called "arguments". Then, and perhaps more

importantly, Structured Argumentation identifies several ways in which the arguments that can be built from a Knowledge Base may be in conflict with one another. These conflicts are called "attacks" and indeed represent where the inconsistencies appear in the Knowledge Base. Yet, Structured Argumentation does not necessarily provide ways to deal with the arguments and attacks it builds when it is done. Which leads us to the second parallel track of research on argumentation in computer science: Abstract Argumentation. Indeed, Abstract Argumentation precisely deals with what to do with arguments and attacks once we have them. And, perhaps surprisingly, this does not classically involve repairing the Knowledge Base from which they might originate. The reason for this is that, since the arguments and attacks are considered to be given, where they come from is not considered important. So, instead, Abstract Argumentation is concerned with which conclusions should be drawn from those that are conflicting. To say it differently, Abstract Argumentation provides ways of selecting arguments based on the conflicts that exist between them. Without delving too much into technicalities, the idea is to select a group of arguments that, together, have some desirable properties regarding the conflicts that are given, and thus can be reasonably chosen as an outcome of these conflicts. This more or less comes down to the general way of deeming that something is reasonable in non-monotonic reasoning: we consider that we can believe in something until provided evidences that it is wrong. In Abstract Argumentation, if we consider conflicts as evidences that arguments are wrong, we thus are to select groups of arguments that are together consistent, and somewhat "resist" these conflicts.

Abstract Argumentation is perfectly interpretable. But, as we more generally discussed previously, this does not necessarily make it understandable for anyone. As we previously discussed, Abstract Argumentation mostly provides selection methods, and indeed, it is not because it is interpretable that a particular selection becomes obvious to anyone. With this comes a problem: if Abstract Argumentation is used to produce explanations for other AI methods, and if Abstract Argumentation is not necessarily understandable for anyone, how can anyone trust its explanations, without further explanations? This observation unveils a major risk for the field of XAI: if we seek to explain AI methods (which can certainly be assumed to not be understandable for anyone) using more AI methods, then we might just end up in a non-ending loop of need for explanations. As such, it is fundamentally critical that this potential loop is broken as soon as possible. Abstract Argumentation has shown its potential to produce explanations for a wide range of AI methods. Yet, it is still an AI method itself, so explanations for the argumentative process are relevant and needed. Therefore, it is critical that the explanations defined for Abstract Argumentation do not rely on more AI methods.

This is *precisely* the subject of the present work: we seek to propose explanations for Abstract Argumentation under the critical constraint that these explanations should be understandable and usable by anyone, even non experts. Since the core utility of Abstract Argumentation is selecting arguments, this is exactly what we will provide explanations for: the validity or non validity of a given selection. We will see that the explanations originally designed for that matter can serve to explain other matters. Additionally, to raise the curtain a little, we propose that our explanations can be understandable and usable by anyone because they are founded on visual modalities.

In addition, it should be noted that since the beginning of its study in computer science, argumentation has had close ties with another domain: *formal logic*. Indeed, in Structured Argumentation, the arguments that are produced are simply premises taken from a Knowledge Base, a conclusion that is inferred from them, and the inference of that conclusion from the premises. Well, the inference system used to derive the conclusion from the premises usually takes the form of a formal logic system. Furthermore, the selection processes of Abstract Argumentation were designed from the start as corresponding to other concepts developed in fields related to formal logic, such as Logic Programming.

Fundamentally, formal logic is the study of reasoning. To be more precise, formal logic is aimed at capturing a particular kind of reasoning: the cognitive process of deduction. When using formal logic, we are interested in determining whether a statement can be deemed to be true via a deductive process. Importantly, this is done solely based on the form of this statement in an abstract dedicated language, and not based on its content. Because the language used is abstract (and usually quite restricted), the cognitive notion of deduction can be captured (at least in essence) using what are called inference rules. It is these inference rules that will be used when determining whether a statement is true or not. To avoid challenging

their correctness, and so the correctness of the entire method, they are usually precise and rather simple. The idea is then to consider a statement to be true if it can be derived from other statements considered to be already true (called axioms) using only the inference rules.

Formal logic is seen and used as a powerful formalization tool. Thus, it should come to no surprise that, in the recent explosion of activity on the subjects of explanations, some researchers worked on representing the explanation process in logic. To be fair, these efforts were already being done before the recent resurgence of interest in explanations, when we tried to explain expert systems for instance. Nonetheless, such a resurgence did boost efforts of researchers on the matter. The point of representing the explanation process in formal logic would be to identify its ground axioms, so its fundamental principles. With this, researchers would have solid rules, laws or at least guidelines under which to develop explanations for various formalisms. To the best of our knowledge however, such principles are not yet available.

Still, despite not having the fundamental mechanisms of explanation, there have been definitions of concepts related to the process of explanation. As such, we consider that it could be beneficial to investigate whether our explanations for Abstract Argumentation correspond to these notions or not. Doing so would require to have a logical account of the explanations we define, but also of what they explain, that is to say the selection of arguments. Thus, in the prospect of conducting a study of our explanations from a logical point of view, we will discuss in the present work logical encoding of both the argument selection process and their explanations. Both of these logical encodings have been designed and developed with the idea of facilitating their extension to various additional aspects of Abstract Argumentation, in particular its generalisations.

To summarize a bit our previous discussions, the present work is about making an interpretable system, namely Abstract Argumentation, explainable. That is to say, it is about *producing explanations for Abstract Argumentation* so that anyone, even non experts, may understand and use them. To do so, we design them so that they rely on a visual modality. Additionally, we wish *to study the status of our explanations as such using logical tools*. To this aim, we provide logical encodings of both the argumentative process and the explanations we defined for it. These two encodings are designed so that they are easy to expand.

The present work is organised as follows. In Chapter 2, we present the basic notions that we need to conduct our research. This includes a presentation of Abstract Argumentation, its basic concepts and related problems, several of its extensions and some particular aspects. Then, in Chapter 3, we present our efforts in defining explanations for Abstract Argumentation that match the constraints we discussed previously. After defining our explanations, we present results as to how to use them and additional results that give insights on their general behavior. We then see how these explanations can be used as a base to build other explanations. In Chapter 4, we present a logical encoding for Abstract Argumentation aimed at capturing its selection mechanisms. This logical encoding is defined in a generic way, in that it can be used for basic Abstract Argumentation or for its extensions by playing on some parameters. In Chapter 5, we then present the logical encoding of the explanations presented in Chapter 3. In the spirit of Chapter 4, we made this logical encoding generic by being able to capture different kinds of explanations by playing on some parameters. We also directed our efforts towards making a logical encoding that as similar as possible to the one presented in Chapter 4, so that the exploration of their links would be easier. Finally, in Chapter 6, we summarize our contributions and findings.

Chapter 2

Preliminary Notions: Abstract Argumentation

2.1 Argumentation Frameworks (AF)

Argumentation Frameworks were first defined by Dung in [Dun95]. They aim at representing abstract entities called "arguments" and the conflicts that emerge between them. These notions are grouped into a directed graph.

Definition 1. An *Argumentation Framework (AF)* is an ordered pair $(\mathscr{A}, \mathscr{R})$ such that $\mathscr{R} \subseteq \mathscr{A} \times \mathscr{A}$.

Vocabulary. For an Argumentation Framework $(\mathscr{A}, \mathscr{R})$, the elements of $\mathscr{A}$ are called *arguments*. $\mathscr{R}$ is an *attack* relation: for $a, b \in \mathscr{A}$, $a\mathscr{R}b$ means that a *attacks* b. This can be extended to sets of arguments: for $S \subseteq \mathscr{A}$ and $b \in \mathscr{A}$, we say that S attacks b if and only if $a\mathscr{R}b$ for some $a \in S$.

Vocabulary. We collectively refer to arguments and attacks as *elements*.

Note. In the present work, we are only interested in *finite* Argumentation Frameworks, that is to say, Argumentation Frameworks where A is finite.

Example. Figures 2.1, 2.2 and 2.3 depict Argumentation Frameworks that we will use throughout this work. In Figure 2.1, a, b, c, d, e are the arguments. We can say that b attacks a and c. We can also say that $\{b, d\}$ attacks c, but not $\{a, d\}$.

We can think of an Argumentation Framework as modeling some debate. This debate can be between several protagonists, or an internal one. Once we have an Argumentation Framework at hand, what we want to achieve is to find the arguments that can collectively "win the debate". Of course, this notion of "winning" may very well vary from one person to another. Nevertheless, in the end, the idea is obtain one (or more) set(s) of arguments with some desirable properties. In his paper, Dung proposed several ways to select such sets of arguments. These sets are called *extensions*, and the ways they are selected *semantics*.

The semantics that Dung defined are inherently tied to the notion of *defence*. The idea is to have something to say against everything that hinders our point of view. To quote directly from Dung's introduction:

> *The way humans argue is based on a very simple principle which is summarized succinctly by an old saying: "The one who has the last word laughs best".*
>
> — Phan Minh Dung, [Dun95]

This way, if the idea is respected, no matter what could be said against the point of view we hold, we would always have something to say back, and thus have "the last word". If we consider that "to have something to say against" means "to attack", we naturally see that to defend an argument, we must attack its attacker.

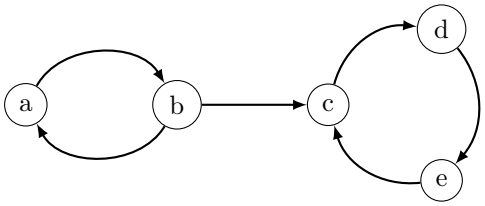

Figure 2.1: A first example of Argumentation Framework
Legend: the nodes are the arguments and the arrows are the attacks

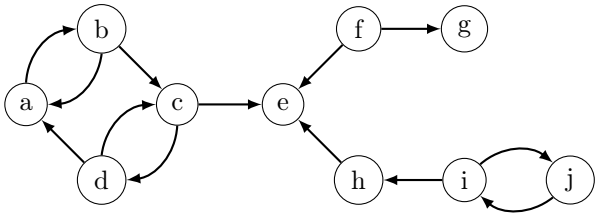

Figure 2.2: A second example of Argumentation Framework

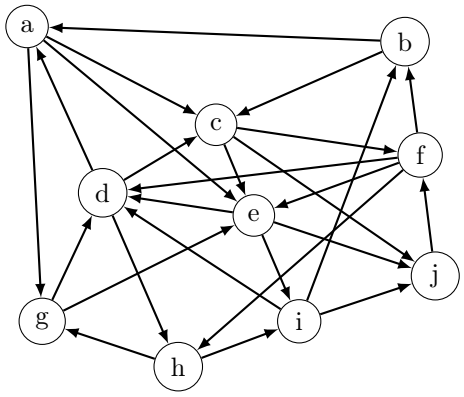

Figure 2.3: A third example of Argumentation Framework

Vocabulary. For an Argumentation Framework $(\mathscr{A}, \mathscr{R})$ and $a, c \in \mathscr{A}$, we say that a *defends* c if and only if $a\mathscr{R}b$ for some $b \in \mathscr{A}$ such that $b\mathscr{R}c$. This can be extended to sets of arguments: for $S \subseteq \mathscr{A}$ and $c \in \mathscr{A}$, we say that S defends c if and only if $a\mathscr{R}b$ for some $a \in S$ and some $b \in \mathscr{A}$ such that $b\mathscr{R}c$.[1]

Example. Back to Figure 2.1, since b attacks c and a attacks b, we can say that a defends c. We can also say that $\{a, b\}$ defends d.

Yet, recall that the initial idea is to have something to say against *everything* that hinders our point of view. As such, it may very well happen that some argument, or set of arguments, defends another argument, in the sense that we just proposed, without it to be enough to correspond to the idea.

Example. In Figure 2.1, although a defends c because it attacks b, it does not attack its second attacker, e. As such, if we consider the point of view made of the two arguments $\{a, c\}$, we might consider it "weak" since there exists a counterargument (e) that none of the arguments composing the point of view can defeat.

This observation leads to the stronger, and fundamentally central, notion of *acceptability*.

Definition 2. Let $(\mathscr{A}, \mathscr{R})$ be an AF. An argument $a \in \mathscr{A}$ is *acceptable* with respect to $S \subseteq \mathscr{A}$ if and only if for all $b \in \mathscr{A}$, if $b\mathscr{R}a$ then $c\mathscr{R}b$ for some $c \in S$.

As such, acceptability is akin to an *effective* defence, a defence that works.

Example. In Figure 2.1, c is not acceptable with respect to $\{a\}$. However, d is acceptable with respect to $\{b\}$.

From the notion of acceptability, we define a function that assigns to a set of arguments the set of arguments that are acceptable with respect to it. We call this function the *characteristic function*.

Definition 3. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an AF. The *characteristic function* of $\mathcal{A}$ is $F_{\mathcal{A}} : 2^{\mathscr{A}} \to 2^{\mathscr{A}}$ such that $F_{\mathcal{A}}(S) = \{a \in \mathscr{A} \mid a \text{ is acceptable with respect to } S\}$ for all $S \subseteq \mathscr{A}$.

With these tools, we can now delve into the ways of selecting arguments in an Argumentation Framework. The semantics originally defined in [Dun95] are as follows.

Definition 4. Let $(\mathscr{A}, \mathscr{R})$ be an AF. A subset S of $\mathscr{A}$ is said to be:

- *conflict-free* iff there are no a and b in S such that a attacks b,
- *admissible* iff S is conflict-free and for all $a \in S$, a is acceptable with respect to S,
- *complete* iff S is admissible and for all $a \in \mathscr{A}$, if a is acceptable with respect to S then $a \in S$,
- *preferred* iff S is maximally (in the sense of set inclusion) admissible,[2]
- *grounded* iff S is the least fixpoint for $F_{\mathcal{A}}$,
- *stable* iff S is conflict-free and S attacks all $a \in \mathscr{A} \setminus S$.

We can think of conflict-freeness as some notion of *coherence*. An admissible extension is thus coherent, and defends all its arguments (it is *self-sufficient*). To use the words of Dung, completeness captures a kind of *confident rationality*, in which one believes in every thing they can defend. The preferred semantics would thus be some *credulous rationality*, in which want to be able to defend our point of view but also accept as much as possible. At the opposite, the grounded semantics would represent some *skeptical rationality*, in which we believe only in what cannot be denied and what follows from it. Finally, the stable semantics represents a *dominant* point of view, in which we want to have something to say against anything.

Example. Tables 2.1, 2.2 and 2.3 give the results of the semantics (omitting conflict-freeness) in the Argumentation Frameworks of Figures 2.1, 2.2 and 2.3 respectively.

[1] In the literature the term "defence" is sometimes associated with the stronger notion of "acceptability" from Definition 2. Here, we dissociate the two so that *being defended* and *being acceptable* have two different meanings.

[2] We write $\subseteq$-*maximal* (or $\subseteq$-*minimal*).

	Admissible	Complete	Preferred	Grounded	Stable
$\varnothing$	✓	✓		✓	
$\{a\}$	✓	✓	✓		
$\{b\}$	✓				
$\{b, d\}$	✓	✓	✓		✓

Table 2.1: Acceptable sets of the Argumentation Framework of Figure 2.1 under the different semantics

	Admissible	Complete	Preferred	Grounded	Stable
$\varnothing$	✓				
$\{b\}$	✓				
$\{d\}$	✓				
$\{f\}$	✓	✓		✓	
$\{i\}$	✓				
$\{j\}$	✓				
$\{a, c\}$	✓				
$\{b, d\}$	✓				
$\{b, f\}$	✓				
$\{b, i\}$	✓				
$\{b, j\}$	✓				
$\{d, f\}$	✓				
$\{d, i\}$	✓				
$\{d, j\}$	✓				
$\{f, i\}$	✓	✓			
$\{f, j\}$	✓				
$\{h, j\}$	✓				
$\{a, c, f\}$	✓	✓			
$\{a, c, i\}$	✓				
$\{a, c, j\}$	✓				
$\{b, d, f\}$	✓	✓			
$\{b, d, i\}$	✓				
$\{b, d, j\}$	✓				
$\{b, f, i\}$	✓				
$\{b, f, j\}$	✓				
$\{b, h, j\}$	✓				
$\{d, f, i\}$	✓				
$\{d, f, j\}$	✓				
$\{d, h, j\}$	✓				
$\{f, h, j\}$	✓	✓			
$\{a, c, f, i\}$	✓	✓	✓		✓
$\{a, c, f, j\}$	✓				
$\{a, c, h, j\}$	✓				
$\{b, d, f, i\}$	✓	✓	✓		✓
$\{b, d, f, j\}$	✓				
$\{b, d, h, j\}$	✓				
$\{b, f, h, j\}$	✓				
$\{d, f, h, j\}$	✓				
$\{a, c, f, h, j\}$	✓	✓	✓		✓
$\{b, d, f, h, j\}$	✓	✓	✓		✓

Table 2.2: Acceptable sets of the Argumentation Framework of Figure 2.2 under the different semantics

	Admissible	Complete	Preferred	Grounded	Stable
$\varnothing$	✓	✓		✓	
$\{a, i, f\}$	✓	✓	✓		✓

Table 2.3: Acceptable sets of the Argumentation Framework of Figure 2.3 under the different semantics

Some properties have been proven in [Dun95] establishing a link between the different semantics. For instance:

Proposition 1. *Given an AF* $(\mathscr{A}, \mathscr{R})$*,*

- *There exists at least one preferred extension.*
- *Every preferred extension is complete, but not vice-versa.*
- *Every stable extension is preferred, but not vice-versa.*
- *The grounded extension is the* $\subseteq$-minimal *complete extension.*

2.2 Classical Problems

In this section, we mention some classical problems that are tackled in Abstract Argumentation. As we mentioned in the previous section, the main mechanism in Abstract Argumentation is the concept of semantics, that is to say, the selection of groups of arguments (extensions). As such, most classical problems revolve around the notion of extension.

Note. All the problems mentioned here are *decision problems*.

Note. Notations and results in this section are taken from [DD18], in which these notions are discussed more deeply. In particular, we will assume the reader familiar with the Computational Complexity Theory.

The first problem that we mention is a very natural one. It consists in verifying whether a given set of arguments is an extension of a given semantics considering a given Argumentation Framework. We call it the *Verification Problem* and we denote it with Ver_σ for a semantics σ.

Ver_σ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ, a set of arguments $S \subseteq \mathscr{A}$
Output : YES if S is an extension of σ in $\mathcal{A}$, NO otherwise

Example. Consider the Argumentation Framework of Figure 2.1, the complete semantics Co and the sets of arguments $\{a\}$ and $\{b\}$. According to Table 2.1, the output of Ver_{Co} on $\{a\}$ is YES, while it is NO on $\{b\}$.

The next problems are tied with the computation of extensions. In some situations, an argumentation may not provide any extension for a given semantics. Verifying whether this occurs can be done easily by trying to compute one extension of the considered semantics (in the case of stable semantics), or one extension different from the empty set in the case of the other semantics. We will call the first problem the *Extension Existence Problem* and denote it $Exists_\sigma$, while we will call the second problem *Non-Empty Extension Existence Problem* and denote it $Exists_\sigma^{\neg\varnothing}$.

$Exists_\sigma$ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ
Output : YES if there exists a set $S \subseteq \mathscr{A}$ such that S is an extension of σ, NO otherwise

$Exists_\sigma^{\neg\varnothing}$ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ
Output : YES if there exists a set $S \subseteq \mathscr{A}$ such that $S \neq \varnothing$ and S is an extension of σ, NO otherwise

Example. Consider the Argumentation Framework of Figure 2.1, the stable semantics *Sta* and the grounded semantics *Gr*. According to Table 2.1, the output of $Exists_{Sta}$ and $Exists_{Gr}$ is YES in both cases, while the output of $Exists^{\neg\varnothing}_{Sta}$ is YES and the output of $Exists^{\neg\varnothing}_{Gr}$ is NO.

Finally, we consider two problems that are related to one another. Indeed, as we can see on Tables 2.1, 2.2 and 2.3, there can be several, and in fact a lot, of different extensions for a fixed semantics. In a situation where we want to make a decision using Abstract Argumentation and we have identified the semantics that would correspond to our decision, this means that we would still have a choice to make between the different possible extensions yielded by that semantics. This is unfortunate as we could have naturally wished to rely on the argumentative process to make that choice for us, that is to say, to identify only one extension.[3]

In that regard, it seems natural to turn to some ways of selecting extensions among the several possible ones for a given semantics. One way to do so is to use what we call the *credulous acceptance* or *skeptical acceptance* of arguments. On the one hand, an argument is *credulously accepted* with respect to some semantics if it belongs to *at least one* extension of that semantics. On the other hand, an argument is *skeptically accepted* with respect to some semantics if it belongs to *every* extension of that semantics.

Credulous and skeptical acceptance can be thought of as making conclusions on arguments at a sort of "global" scale of the Argumentation Framework. They can be used to filter out a bit the extensions we obtain via the Enumeration Problem. For instance, if a particular argument is credulously accepted, we might want to keep only the extensions that contain it. On the contrary, if an argument is skeptically accepted, we cannot try to keep extensions that do not contain it. We can define associated problems that consist in verifying whether a given argument is credulously accepted or not, or skeptically accepted of not.

$Cred_\sigma$ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ, an argument $a \in \mathscr{A}$
Output : YES if a belongs to at least one extension of σ in $\mathcal{A}$, NO otherwise

$Skept_\sigma$ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ, an argument $a \in \mathscr{A}$
Output : YES if a belongs to every extension of σ in $\mathcal{A}$, NO otherwise

Example. Consider the Argumentation Framework of Figure 2.1, the preferred semantics *Pr* and the arguments *a* and *b*. According to Table 2.1, the output of $Cred_{Pr}$ is YES, while the output of $Skept_{Pr}$ is NO, both for *a* and *b*.

We finish by summarizing in Table 2.4 the complexity of solving each problem mentioned in this section, for each semantics mentioned in Definition 4. This is a reduced version of the similar Table given in [DD18].

2.3 Enrichments for Argumentation Frameworks

Here, we will detail several generalisations of Argumentation Frameworks that we call *enrichments*. Some of them have already been given names, however we wish to adopt a more functional naming process in

[3]Note that the grounded semantics has been proved to yield only one extension, but this extension is empty if all arguments are attacked by another argument.

σ	Ver_σ	$Exists_\sigma$	$Exists^{\neg\varnothing}_\sigma$	$Cred_\sigma$	$Skept_\sigma$
CF	in L	Trivial	in L	in L	Trivial
Adm	in L	Trivial	NP-c	NP-c	Trivial
Co	in L	Trivial	NP-c	NP-c	P-c
Gr	P-c	Trivial	in L	P-c	P-c
Pr	coNP-c	Trivial	NP-c	NP-c	Π^P_2-c
Sta	in L	NP-c	NP-c	NP-c	coNP-c

Table 2.4: Complexity of classical problems in Abstract Argumentation in function of the semantics ($\mathcal{C}$-c denotes completeness for complexity class $\mathcal{C}$)

this work. Accordingly we will precise how a given generalisation was named in the literature, and how we will call it in the present work. We will present them in isolation at first, and then define a Generalised Argumentation Framework that captures them all together.

For readability reasons, arcs will be identified by names, as it is classically done for vertices. We will use Latin letters to identify arguments and Greek letters to identify arcs.

2.3.1 Argumentation Frameworks with Coalitions

The formalism we call Argumentation Framework with Coalitions (AF-C) is called SETAF (for "Framework with sets of attacking arguments") in the literature, and has been studied for instance in [NP06], [FB19].

In an Argumentation Framework with Coalitions, we consider that the arguments may not only attack each other individually, but may also do so in groups. That is to say, attacks may originate from *sets of arguments*, and may also target *sets of arguments*. As such, Argumentation Frameworks with Coalitions are just the generalisation of Argumentation Frameworks to hypergraphs. In the present work however, we will restrict ourselves to the particular case of attacks targeting only *singleton sets*. In other words, attacks continue to target individual arguments only.

As a general discussion before going to definitions, this generalisation raises of course the question of how to interpret the attack relation. Typically, when is an attack effective? What effect does it have? As it is often the case when sets are involved, the new interpretation relies on the dual notions of *universality* and *existentiality*: either all or (at least) one element of the set is affected. Thus, an attack that originates from a set of arguments would be effective when either all the arguments of the set, or at least one, satisfy a particular condition. Likewise, an attack targeting a set of arguments could enforce a particular condition on either all arguments of the set, or at least one. In reality, the matter is actually more complicated. The meaning of a group of arguments being attacked together is a controversial subject, without any emerging consensus. The reader can see [DDK+23] for a short overview of possibilities.

The choices made between the different possibilities eventually lead to different interpretations. In the present work, we will consider that an attack, in such a setting, is effective when *all the arguments* of the set it originates from satisfy a particular condition.

Definition 5. An AF-C is a tuple $(\mathscr{A}, \mathscr{R}, s, t)$ where $\mathscr{A}$ is a set of arguments and $\mathscr{R}$ is a set of attacks such that:

- $\mathscr{A} \cap \mathscr{R} = \varnothing$,
- $s : \mathscr{R} \to 2^{\mathscr{A}} \setminus \{\varnothing\}$,
- $t : \mathscr{R} \to \mathscr{A}$.

Vocabulary. For an Argumentation Framework with Coalitions $(\mathscr{A}, \mathscr{R}, s, t)$ and an attack $\alpha \in \mathscr{R}$, $s(\alpha)$ is called the *source* of α and $t(\alpha)$ is called the *target* of α.

Vocabulary. For an Argumentation Framework with Coalitions $(\mathscr{A}, \mathscr{R}, s, t)$, $S \subseteq \mathscr{A}$ and $b \in \mathscr{A}$, we say that S attacks b if and only if there is some $\alpha \in \mathscr{R}$ such that α targets b and has a subset of S as its source.

To retrieve the notion of defence, recall our choice of interpretation: all the arguments from the source of an attack need to be viable for the attack to be effective. Thus, if one of them is challenged, the entire attack is crippled. As such, an argument needs only to see *one* argument from the source of one of the attacks that target it attacked back.

Vocabulary. For an Argumentation Framework with Coalitions $(\mathscr{A}, \mathscr{R}, s, t)$, $S \subseteq \mathscr{A}$ and $c \in \mathscr{A}$, we say that S defends c if and only if for some $\alpha \in \mathscr{R}$ such that $t(\alpha) = c$, $\exists \beta \in \mathscr{R}$ with $t(\beta) \in s(\alpha)$ and $s(\beta) \subseteq S$.

Example. Figure 2.4 depicts an Argumentation Framework with Coalitions, derived from the Argumentation Framework of Figure 2.1. In this context, we can say that $\{a, f\}$ attacks b. $\{a, f\}$ is the source of α, while b is the target of α. As $\{b\}$ attacks c, we can say that $\{a, f\}$ defends c.

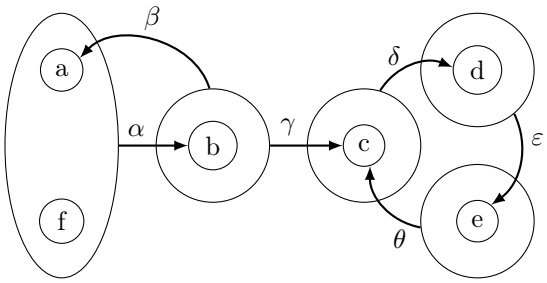

Figure 2.4: An example of an Argumentation Framework with Coalitions

	Admissible	Complete	Preferred	Grounded	Stable
$\varnothing$	✓				
$\{b\}$	✓				
$\{f\}$	✓	✓		✓	
$\{a, f\}$	✓	✓	✓		
$\{b, d\}$	✓				
$\{b, f\}$	✓				
$\{b, d, f\}$	✓	✓	✓		✓

Table 2.5: Acceptable sets of the Argumentation Framework with Coalitions of Figure 2.4 under the different semantics

We now provide definitions for the classical semantics for Argumentation Frameworks with Coalitions. As it turns out, we need not repeat definitions beyond acceptability and conflict-freeness, as they straightforwardly carry over from Argumentation Frameworks.

Definition 6. Let $(\mathscr{A}, \mathscr{R}, s, t)$ be an AF-C. An argument $a \in \mathscr{A}$ is *acceptable* with respect to $S \subseteq \mathscr{A}$ if and only if $\forall \alpha \in \mathscr{R}$ such that $t(\alpha) = a$, $\exists \beta \in \mathscr{R}$ with $t(\beta) \in s(\alpha)$ and $s(\beta) \subseteq S$.

Definition 7. Let $(\mathscr{A}, \mathscr{R}, s, t)$ be an AF-C. A subset S of $\mathscr{A}$ is said to be *conflict-free* if and only if $\nexists b \in S$ and $S' \subseteq S$ such that $\exists \alpha \in \mathscr{R}$ where $s(\alpha) = S'$ and $t(\alpha) = b$.

Example. Table 2.5 gives the results of the semantics (omitting conflict-freeness) in the Argumentation Framework with Coalitions of Figure 2.4.

2.3.2 Higher-Order Argumentation Frameworks

The formalism we call Higher-Order Argumentation Framework (HO-AF) is a generalisation that also leads to several interpretations. However, at the opposite of Argumentation Frameworks with Coalitions, in the literature, one name was given for each interpretation. This generalisation was introduced in [BGW05], and then developed in several papers, among which one can cite the AFRA (Argumentation Framework with Recursive Attacks) interpretation, described in [BCGG11], and the RAF (Recursive Argumentation Framework) interpretation described in [CFFL21]. Note that despite the different names and the different interpretations, there is no difference in the structure of the graph.

Roughly speaking, Higher-Order Argumentation Frameworks are named so because they introduce higher-order attacks, that is to say attacks that may *target other attacks* as well as arguments (hence the idea of *recursive* attacks in the names of the literature). In the present work, we are only interested in the **RAF interpretation** of the generalisation, leaving the AFRA interpretation to future considerations.

Definition 8. An HO-AF is a tuple $(\mathscr{A}, \mathscr{R}, s, t)$ where $\mathscr{A}$ is a set of arguments and $\mathscr{R}$ is a set of attacks such that:

- $\mathscr{A} \cap \mathscr{R} = \varnothing$,
- $s : \mathscr{R} \to \mathscr{A}$,
- $t : \mathscr{R} \to \mathscr{A} \cup \mathscr{R}$.

Vocabulary. Just as with AF-C, s returns the *source* of an attack and t returns its *target*.

To retrieve the notion of attack, recall that it is now possible for an attack to target another attack instead of an argument. So, the notion of being attacked is to be extended to all elements of $\mathscr{A} \cup \mathscr{R}$.

Vocabulary. For a Higher-Order Argumentation Framework $(\mathscr{A}, \mathscr{R}, s, t)$, $a \in \mathscr{A}$ and $x \in \mathscr{A} \cup \mathscr{R}$, we say that a *attacks* x if and only if there is some $\alpha \in \mathscr{R}$ such that α targets x and has a as its source. This can be extended to sets of arguments: for $S \subseteq \mathscr{A}$ and $x \in \mathscr{A} \cup \mathscr{R}$, we say that S attacks x if and only if a attacks x for some $a \in S$.

The adaptation of the notion of defence is where lies the real fundamental difference between the AFRA interpretation and the RAF interpretation. In an Argumentation Framework, to defend an argument is to attack one of its attackers. Thus, attacking the source of an attack suffices to take care of the attack. Now, in our setting, attacks can be directly targeted as well, which should provide another way of dealing with them. And indeed, both in the AFRA and the RAF interpretations, directly attacking an attack or attacking its source are the two ways of neutralizing that attack. However, this also raises the new questions of how and when to defend attacks. We might consider that we need to defend an attack when either it or its source is attacked. This is indeed the choice that is made in the AFRA interpretation. However, in the RAF interpretation, we only need to defend an attack when it is *directly attacked*, and not necessarily when its *source is attacked*.

Vocabulary. For a Higher-Order Argumentation Framework $(\mathscr{A}, \mathscr{R}, s, t)$, $a \in \mathscr{A}$ and $x \in \mathscr{A} \cup \mathscr{R}$, we say that a *defends* x if and only if a attacks α or $s(\alpha)$ for some $\alpha \in \mathscr{R}$ such that $t(\alpha) = x$. This can be extended to sets of arguments: for $S \subseteq \mathscr{A}$ and $x \in \mathscr{A} \cup \mathscr{R}$, we say that S defends x if and only if a attacks α or $s(\alpha)$ for some $a \in S$ and for some $\alpha \in \mathscr{R}$ such that $t(\alpha) = x$.

Example. Figure 2.5 depicts a Higher-Order Argumentation Framework, derived from the Argumentation Framework of Figure 2.1. In this context, we can say that a attacks b, but also that g attacks γ. a is the source of α, while b is the target of α. As $\{b\}$ attacks c, we can say that a defends c, but as γ is the attack targeting c, we may also say that g defends c. Additionally, we can say that $\{a, g\}$ attacks b and defends c.

We now provide the basic notions needed to express the classical semantics for a Higher-Order Argumentation Framework. Since all elements of $\mathscr{A} \cup \mathscr{R}$ can be attacked, the notions of acceptability and conflict-freeness are relative to both a subset of $\mathscr{A}$ and a subset of $\mathscr{R}$. In the RAF interpretation, we tend to consider these subsets separately, while in the AFRA interpretation, we tend to consider the union of these sets. Since the present work assumes arguments to be disjoint from attacks, this does not change much for us.

The difference between the two interpretations in the notion of defence leads to different results for some semantics, although not all of them.

Definition 9. Let $(\mathscr{A}, \mathscr{R}, s, t)$ be an HO-AF. An element $x \in \mathscr{A} \cup \mathscr{R}$ is *acceptable* with respect to $S \subseteq \mathscr{A}$ and $\Gamma \subseteq \mathscr{R}$ if and only if $\forall \alpha \in \mathscr{R}$, if $t(\alpha) = x$, then there exists $\beta \in \Gamma$ such that $s(\beta) \in S$ and either $t(\beta) = \alpha$ or $t(\beta) = s(\alpha)$.

Definition 10. Let $(\mathscr{A}, \mathscr{R}, s, t)$ be an HO-AF. For $S \subseteq \mathscr{A}$ and $\Gamma \subseteq \mathscr{R}$, (S, Γ) is said to be *conflict-free* if and only if $\forall \alpha \in \Gamma$ such that $s(\alpha) \in S$, $t(\alpha) \notin S \cup \Gamma$.

As for Argumentation Frameworks with Coalitions, the definitions beyond acceptability and conflict-freeness straightforwardly carry from Argumentation Frameworks, provided we define them for pairs (S, Γ).

Example. Table 2.6 gives the results of the semantics (omitting conflict-freeness and admissibility) in the Higher-Order Argumentation Framework Figure 2.5.

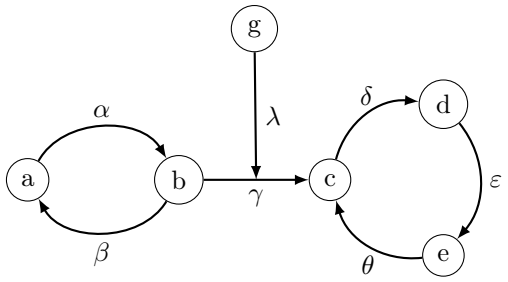

Figure 2.5: An example of a Higher-Order Argumentation Framework

	Complete	Preferred	Grounded	Stable
$(\{g\}, \{\alpha, \beta, \delta, \epsilon, \theta, \lambda\})$	✓		✓	
$(\{a, g\}, \{\alpha, \beta, \delta, \epsilon, \theta, \lambda\})$	✓	✓		
$(\{b, g\}, \{\alpha, \beta, \delta, \epsilon, \theta, \lambda\})$	✓	✓		

Table 2.6: Acceptable sets of the Higher-Order Argumentation Framework of Figure 2.5 under the different semantics

Note. Please note, regarding admissibility, that there are in fact a very large number of admissible extensions. Indeed, $(\{a\}, \{\alpha\})$ is admissible for instance, but we can also add any number of additional attacks (except γ) to this extension and obtain a new admissible extension. Regarding definitions 9 and 10, this is due to the fact that all attacks (except γ) are unattacked themselves, and that a is only the source of α. This leads to an exponentially large number of admissible extensions with respect to the number of both attacks and arguments, and not just arguments.

2.3.3 Bipolar Argumentation Frameworks

The formalism we call Bipolar Argumentation Framework (BAF) is a generalisation in which we consider an additional binary relation between the arguments: the relation of *support*. The support relation is generally considered to be a "positive" relation between arguments, at the opposite of the attack relation. This enrichment was introduced in [KP01, CL05]. Just like the previous enrichments, there are several interpretations of the support relation. [CL13] gives a comparative study of these different interpretations. As with Higher-Order Argumentation Frameworks, although the different possible interpretations rely on different intuitions, the structure of the graph is the same for all of them.

Definition 11. A BAF is a tuple $(\mathscr{A}, \mathscr{R}, \mathscr{S}, s, t)$ where $\mathscr{A}$ is a set of arguments, $\mathscr{R}$ is a set of attacks and $\mathscr{S}$ is a set of supports such that:

- $\mathscr{A} \cap \mathscr{R} = \mathscr{A} \cap \mathscr{S} = \mathscr{R} \cap \mathscr{S} = \varnothing$,
- $s : \mathscr{R} \cup \mathscr{S} \to \mathscr{A}$,
- $t : \mathscr{R} \cup \mathscr{S} \to \mathscr{A}$.

Vocabulary. We collectively refer to attacks and supports as *interactions*, and to arguments, attacks and supports as *elements*.

Vocabulary. Like with the previous enrichments, s returns the *source* of an interaction and t returns its *target*.

In Argumentation Frameworks, the attack relation is interpreted in a somewhat "immediate scope": *attacking* an argument means to have an attack arc directed to it. Additionally, attacking an attacker is considered *defence*. Thus, along a path in the attack relation ending in a fixed argument, the individual

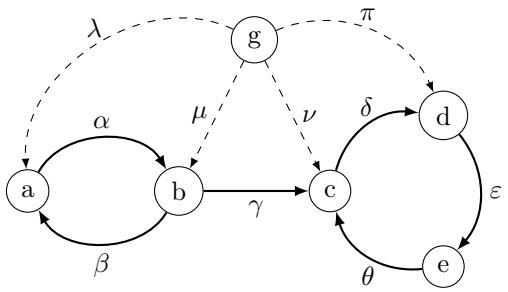

Figure 2.6: An example of a Bipolar Argumentation Framework
Legend: The plain arrows are the attacks and the dashed arrows are the supports

meaning of the arcs fluctuates between attack and defence (regarding the last, fixed, argument). This is not the case for the support relation. Its meaning is sort of transferred from arguments to arguments via such a chain, remaining constant.

Vocabulary. For a Bipolar Argumentation Framework $(\mathscr{A}, \mathscr{R}, \mathscr{S}, s, t)$ and $a, b \in \mathscr{A}$, we say that *a supports b* if and only if there is a path from a to b in the support relation.

Note. Notice the difference of behavior between attacks and supports. Attacks could be said to be interpreted "immediately" or "locally". Their scope is limited to their target. That is why, along a path in the attack relation, starting from a fixed argument, the effect of successive attacks will switch between offense and defence. On the contrary, for the support relation, the same interpretation is preserved and propagated along a given path. Thus, supports could be said to be interpreted "globally".

In our figures, we will differentiate supports from attacks by depicting supports as dashed arrows.

Example. Figure 2.6 depicts a Bipolar Argumentation Framework, derived from the Argumentation Framework of Figure 2.1. In this context, α is an attack and λ a support. Both are interactions. We can say that g supports a, b, c and d.

As we have already said, there are several possible interpretation of the support relation. We now present three of them, which are all studied in [CL13]. In this work, we will focus on only one out the three.

The *deductive support* interpretation has been introduced in [BGvdTV10]. The main idea behind this interpretation is to consider that, for an argument a that supports an argument b, if a is selected in an extension, then b should be as well (because b can be *deduced* from a). The *necessary support* interpretation comes from [NR11]. In this interpretation, we consider that, for an argument a that supports an argument b, if b is selected in an extension, then a should be as well (because a is *necessary* for b). In both of these interpretations, the effect of the support relation is taken into account via the addition of new attacks, on the basis of the supports. Interestingly, these two interpretations are dual: the results in a Bipolar Argumentation Framework with deductive support are identical to those in the same graph, in which the direction of the support arcs has been reversed, interpreted as having a necessary support.

The interpretation we focus on in the present work is the *evidential support* interpretation. It was first defined in [ON08], then revised in [PO14]. The idea behind this interpretation is that arguments, to be deemed worth of consideration, should receive support from particular arguments that we call *prima-facie*. Prima-facie arguments can be thought of as some sort of undeniable truth, much like *evidences* (hence the name of the interpretation). A Bipolar Argumentation Framework with evidential support is called Evidence-Based Argumentation Framework (EBAF) in the literature, a name we shall keep in the present work. An Evidence-Based Argumentation Framework is merely a Bipolar Argumentation Framework with an additional set containing the prima-facie arguments.

Note. In [ON08] and [PO14], Evidence-Based Argumentation Framework are defined with interactions that can have sets of arguments as their source. In order to deal with enrichments separately, we provide a definition where only a single argument can be the source of an interaction.

Definition 12. An EBAF is a tuple $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ where $\mathscr{A}$ is a set of arguments, $\mathscr{R}$ is a set of attacks and $\mathscr{S}$ is a set of supports such that:

- $\mathscr{A} \cap \mathscr{R} = \mathscr{A} \cap \mathscr{S} = \mathscr{R} \cap \mathscr{S} = \varnothing$,
- $\mathscr{P} \subseteq \mathscr{A}$,
- $s : \mathscr{R} \cup \mathscr{S} \to \mathscr{A}$,
- $t : \mathscr{R} \cup \mathscr{S} \to \mathscr{A}$.

Vocabulary. For an Evidence-Based Argumentation Framework $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ and $a \in \mathscr{A}$, we say that a is *prima-facie* if and only if $a \in \mathscr{P}$.

In our figures, we will differentiate prima-facie arguments from regular arguments by depicting them as double circled.

Of course, semantics in an Evidenced-Based Argumentation Framework must take into account the support relation. The following definitions capture this need for support and serve as the basis of more complex ones.

Definition 13. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. An argument $a \in \mathscr{A}$ is *e-supported* if and only if $a \in \mathscr{P}$ or there exists $b \in \mathscr{P}$ such that b supports a.[4]

Note. Notice the difference between the notion of support as previously given in vocabulary and this notion of e-support. Support is defined regarding two arguments, while e-support is defined regarding only one. As such, e-support is more akin to a notion of "global" support, at the scale of the entire framework.

Note. The notion of e-support can be restricted regarding a given set.

Definition 14. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. An argument $a \in \mathscr{A}$ is *e-supported* by a set $S \subseteq \mathscr{A}$ if and only if $a \in \mathscr{P}$ or $\exists \alpha \in \mathscr{S}$ with (1) $t(\alpha) = a$, (2) $s(\alpha) \in S$, and (3) $s(\alpha)$ is *e-supported* by $S \setminus \{a\}$.

Note. Notice the constraint in condition (3): an argument must be supported without itself.

Note. In Definition 14, taking $S = \mathscr{A}$ yields a different (in particular, recursive) but equivalent definition of e-support as defined in Definition 13.

Vocabulary. For an Evidence-Based Argumentation Framework $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ an argument $a \in \mathscr{A}$, and a set of arguments $S \subseteq \mathscr{A}$, we say that a is *minimally e-supported by* S if and only if $\nexists S' \subset S$ such that a is e-supported by S'.

The previous definitions give us the first brick that we need to build the semantics.

Definition 15. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. A set $S \subseteq \mathscr{A}$ is *self-supported* if and only if $\forall a \in S$, a is e-supported by S.

As we said before, the intuition is that only the arguments that are supported (in the sense we said earlier) by a prima-facie argument should be considered. To make sure this occurs, we redefine even the notion of attack.

Definition 16. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. An argument $a \in \mathscr{A}$ *e-attacks* an argument $b \in \mathscr{A}$ if and only if a attacks b and a is e-supported.

Note. The notion of e-attack can be defined regarding a given set.

Definition 17. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. A set $S \subseteq \mathscr{A}$ *e-attacks* an argument $b \in \mathscr{A}$ if and only if there exists $a \in S$ such that a attacks b and a is e-supported by S.

[4] Keep in mind how we said what "b supports a" means earlier: "there is a path from b to a in the support relation".

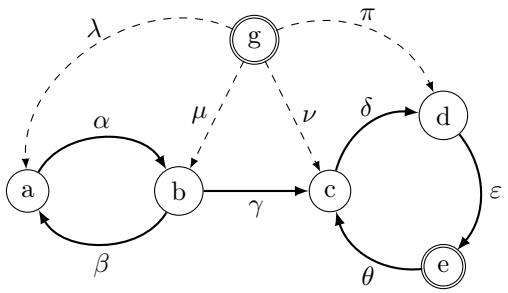

Figure 2.7: An example of an Evidence-Based Argumentation Framework

	Admissible	Complete	Preferred	Grounded	Stable
$\varnothing$	✓				
$\{g\}$	✓	✓		✓	
$\{a, g\}$	✓	✓	✓		
$\{b, g\}$	✓				
$\{b, d, g\}$	✓	✓	✓		✓

Table 2.7: Acceptable sets of the Evidence-Based Argumentation Framework of Figure 2.7 under the different semantics

Note. Definition 17 is neither a *restriction* nor a *generalisation* of Definition 16 because, even if potential attackers are grouped in a set, support (which is essential in this setting) is confined to within this set.

Example. Figure 2.7 depicts an Evidence-Based Argumentation Framework, which is simply the Bipolar Argumentation Framework of Figure 2.6, where e and g are prima-facie arguments. In this context, all the arguments are e-supported. In particular, e and g are e-supported by $\varnothing$ and the other arguments are e-supported by $\{g\}$. We can say that a e-attacks b, however $\{a\}$ does not e-attack b: $\{a, g\}$ does.

We are now ready to define acceptability in the context of an Evidence-Based Argumentation Framework.

Definition 18. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. An argument a is *e-acceptable* with respect to $S \subseteq \mathscr{A}$ if and only if (1) a is e-supported by S and (2) for any $T \subseteq \mathscr{A}$ that e-attacks a, S e-attacks an element of T.

Most of the usual semantics are expressed as in Definition 4 by changing acceptability for e-acceptability. We give only the semantics for which a change occurs.

Note. Notice that, by condition (1) of Definition 18, a set of arguments such that all its arguments are e-acceptable with respect to it is self-supported.

Definition 19. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. A set $S \subseteq \mathscr{A}$ is *conflict-free* if and only if there are no a and b in S such that $\exists \alpha \in \mathscr{R}$ with $s(\alpha) = a$ and $t(\alpha) = b$.

Remark. Please observe that Definition 19 is merely the definition of conflict-freeness from Definition 4 adapted to the use of functions s and t.

Definition 20. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. A set $S \subseteq \mathscr{A}$ is a *stable* extension if and only if (1) S is conflict-free, (2) S is self-supported and (3) $\forall a \in (\mathscr{A} \setminus S)$ such that a is e-supported, S attacks a or S attacks $b \in T$ for any $\subseteq$-*minimal* set T such that a is e-supported by T.

Example. Table 2.7 gives the results of the semantics (omitting conflict-freeness) in the Evidence-Based Argumentation Framework of Figure 2.7.

2.4 Decomposition of Abstract Argumentation semantics

Recall that the entire idea of Abstract Argumentation is to have ways of selecting groups of arguments that possess some desirable properties. We have seen that this idea is captured by the concept of *semantics*. Semantics group together these properties and offer ready-to-use ways of selection arguments. In addition, we can see in Definition 4 that semantics are somewhat built on each other. Consider the following:

- a set of arguments is conflict-free if and only if it has no internal conflicts,
- a set of arguments is admissible if and only if it is conflict-free and respects another property,
- a set of arguments is complete if and only if it is admissible and respects another property.

This way, we see that some of the sort of *atomic conditions* that constitute semantics are shared between several of them, due to how they are defined on one another. As such, people have explored how to decompose Abstract Argumentation semantics into a set of atomic conditions, and redefine them in terms of which atomic conditions must be satisfied to retrieve the semantics. This would allow to have a *modular* understanding of semantics.

Vocabulary. In the following, we will call *underlying principles* (or simply *principles*) the atomic conditions that constitute Abstract Argumentation semantics.

The study of the decomposition of Abstract Argumentation semantics into underlying principles has already been done in [DM16]. Considering an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ and a set of arguments $S \subseteq \mathscr{A}$, the authors identify the following principles:

Coherence (*Coh*):	there exists no internal conflicts in S
Defence (*Def*):	$\forall x \in S$, x is acceptable with respect to S
Reinstatement (*Rein*):	$\forall x$ acceptable with respect to S, $x \in S$
Complement Attack (*CA*):	S attacks all arguments not in S
Maximality (*Max*):	S is $\subseteq$-*maximal*
Minimality (*Min*):	S is $\subseteq$-*minimal*

Note. The Coherence principle is also called the Conflict-freeness principle in the literature, for reasons that we are about to see. In this work, we use a different one to clearly distinguish the moments when we talk about semantics and when we talk about principles.

Note. The Maximality and Minimality principles are to be understood relatively to other principles. The idea is for the set to be $\subseteq$-*maximal* (or $\subseteq$-*minimal*) *such that other principles are respected as well.*

Example. Consider the Argumentation Framework of Figure 2.1. The set $\{a\}$ respects the Coherence and Defence principles. Likewise, the set $\{b, d\}$ respects the Coherence and Complement Attack principles.

Notation. In the following, provided that they are not used to name arcs, we will use the letter σ to denote some Abstract Argumentation *semantics*, and the letter π to denote some Abstract Argumentation *principle*.

Following the identification of underlying principles, a correspondence was established in [DM16] between semantics and respect of some principles. This is the object of the next Proposition:

Proposition 2. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq A$. S is:*

Conflict-free	*if and only if S respects*	*{ Coh }*
Admissible	*if and only if S respects*	*{ Coh, Def }*
Complete	*if and only if S respects*	*{ Coh, Def, Rein }*
Preferred	*if and only if S respects*	*{ Coh, Def, Max }*
Grounded	*if and only if S respects*	*{ Coh, Def, Rein, Min }*
Stable	*if and only if S respects*	*{ Coh, CA }*

Example. Consider the Argumentation Framework of Figure 2.1. We have already seen that $\{a\}$ respects the Coherence and Defence principles, and $\{b, d\}$ respects the Coherence and Complement Attack principles. This makes $\{a\}$ an admissible extension and $\{b, d\}$ a stable extension (verifiable on Table 2.1).

We have seen that Abstract Argumentation semantics can be decomposed into underlying principles in the case of Argumentation Frameworks. We have also seen that Argumentation Frameworks can be be generalised using different enrichments. Considering this, it seems natural to wonder whether the decomposition we have seen still holds when enrichments are added, or whether the addition of enrichments somehow brings new atomic underlying principles to consider. This is what we will look at now.

Coalitions and Higher-Order interactions As it turns out, the addition of coalitions and higher-order interactions does not bring new underlying principles to consider. This means that, in these enriched frameworks, semantics can be decomposed in the same way that we have seen, and Proposition 2 still holds. That is, of course, provided that we use the definitions corresponding to the framework at hand. So, for instance, the Defence principle which states for a set of arguments S "$\forall x \in S$, x is acceptable with respect to S" would then refer to Definition 6 of acceptability in the case of an Argumentation Framework with Coalitions, and to Definition 9 in the case of a Higher-Order Argumentation Framework.

Evidential support In the case of an Evidence-Based Argumentation Framework however, we need to consider an additional underlying principle. This principle corresponds to the intuition that, in this case, arguments should receive support (using the evidential interpretation of "receiving support"). For a set of arguments S, this in fact refers to Definition 15 of *self-support*.

Now, recall that we already observed that, by Definition 18, any set of arguments such that all its arguments are e-acceptable with respect to it is self-supported. Thus, one way to include this new principle could be to change the phrasing of principles that use the notion of acceptability to replace it by e-acceptability. Thus, for example, the Defence principle would become, for a set of arguments S, "$\forall x \in S$, x is e-acceptable with respect to S". This way, Proposition 2 could still be used in the case of Evidence-Based Argumentation Frameworks. Another way to include the new principle would be to add it to the list, and provide a definition for acceptability (which is different from e-acceptability) in Evidence-Based Argumentation Frameworks (it would thus be a softer version of Definition 18 of e-acceptability, not taking into account the aspects of support). This would then require to give an adapted version of Proposition 2 in the case of Evidence-Based Argumentation Frameworks. Since the initial idea of this approach was to decompose semantics into *atomic* principles, and the Defence principle using e-acceptability can be separated into two principles, we choose to do the latter, which we consider to be closer to the intuition of the approach.

We begin with giving a definition for the concept of acceptability for Evidence-Based Argumentation Frameworks. Again, we insist that this notion of acceptability *does not correspond* per se to the notion of e-acceptability from Definition 18. It is meant to be the union of the notions of acceptability and self-support that allows to retrieve e-acceptability. As such, in acceptability, the consideration that the arguments of the extension should be supported by it is left aside.

Definition 21. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. An argument a is *acceptable* with respect to $S \subseteq \mathscr{A}$ if and only if for any $T \subseteq \mathscr{A}$ that e-attacks a, S attacks an element of T.

Remark. Notice that in Definition 21, in addition of removing the condition of a being e-supported by S, we also require that S simply attacks an element of T, instead of using the notion of e-attack. In practice, since this notion is to be supplemented with self-support, the attack from S will indeed be an e-attack.

We now give the new list of principles to use, including the new principle of Self-support. Considering an Evidence-Based Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ and a set of arguments $S \subseteq \mathscr{A}$, we consider the following principles:

Coherence (*Coh*):	there exists no internal conflicts in S
Self-support (*SS*):	$\forall x \in S$, x is e-supported by S
Defence (*Def*):	$\forall x \in S$, x is acceptable[5] with respect to S
Reinstatement (*Rein*):	$\forall x$ acceptable with respect to S, $x \in S$
Complement Attack (*CA*):	$\forall x \notin S$, if x is e-supported, S attacks x or every chain of support of x
Maximality (*Max*):	S is $\subseteq$-*maximal*
Minimality (*Min*):	S is $\subseteq$-*minimal*

Remark. Note that the Complement Attack principle also has to be changed to take supports into account, in correspondence with Definition 20.

Finally, we give an adapted version of Proposition 2 for the case of Evidence-Based Argumentation Frameworks.

Proposition 3. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq A$. S is:*

Conflict-free	*if and only if S respects*	*{ Coh }*
Admissible	*if and only if S respects*	*{ Coh, SS, Def }*
Complete	*if and only if S respects*	*{ Coh, SS, Def, Rein }*
Preferred	*if and only if S respects*	*{ Coh, SS, Def, Max }*
Grounded	*if and only if S respects*	*{ Coh, SS, Def, Rein, Min }*
Stable	*if and only if S respects*	*{ Coh, SS, CA }*

Example. Consider the Evidence-Based Argumentation Framework of Figure 2.7. $\{a, g\}$ respects the Coherence, Self-support and Defence principles. Additionally, $\{b, d, g\}$ respects the Coherence, Self-support and Complement Attack principles. According to Proposition 3, $\{a, g\}$ is then an admissible extension, while $\{b, d, g\}$ is a stable extension, which is verifiable on Table 2.7.

[5]Of course, the notion of "acceptable" here refers to Definition 21

Chapter 3

Visual Explanations for Abstract Argumentation

In this chapter, we are interested in providing explanations for results obtained via a selection process from Abstract Argumentation. In particular, we have the ambition to make these explanations understandable and usable by anyone, even by people that are not familiar with notions of Abstract Argumentation. To do so, we will make efforts to give an important *visual modality* to our explanations. This way, instead of using theoretical notions from Abstract Argumentation to understand and use an explanation, it will be possible to only rely on *how it looks like*, which would indeed make it accessible to anyone.

The chapter is organized as follows: we begin by presenting the relevant related works, by putting the accent on what problems are explanations defined for in the literature, and how they are defined (Section 3.1). Next, in Section 3.2, we motivate the explanation problem that is addressed in the present work, which is different from the main problem addressed in the literature. We also provide some hypotheses on which we rely to tackle the problem we address. Then, in sections 3.4 and 3.5, we formally define our explanations and give formal results on how they can be used, how they are organized and how to compute them. We proceed with showing how these explanations can be used to build new kind of explanations, addressing a different problem. In the end, Section 3.6, we summarize our work briefly by restating the main points of our contribution and giving an example that illustrates the whole approach. Finally, we compare our work to the related works, discuss on how to assess the quality of the explanations we defined and present some ideas for future lines of research (sections 3.7 to 3.9). Note also that we recall in Section 3.3 some useful technical tools.

3.1 Related Works

Before developing our approach, we present several existing works related to the computation of explanations for Abstract Argumentation, and also on explanations in general. For the first part, the presentation is organised following the "taxonomy" of the types of explanation proposed in the survey [ČRA+21]: explanations can be defined as either subgraphs, changes, extensions (sets of arguments), or dialogue-games.

Subgraphs. Let us consider first the category of *subgraphs*. A first example of work defining explanations as *subgraphs* is [SWW20]. It was in fact categorised in the second category (change) in [ČRA+21], a choice which can be discussed. Indeed, [SWW20] seeks to explain the credulous non acceptance of some argument, not by changing its status, but by finding a Strongly Rejecting Subframework. A Strongly Rejecting Subframework is an induced subgraph of an Argumentation Framework that does not credulously accept an argument, and nor do its supergraphs that are still induced subgraphs of the original Argumentation Framework. As such, Strongly Rejecting Subframeworks aim to capture the core argumentative reasons for why an argument is not credulously accepted under a certain semantics.

[NJ20] also studies subgraphs to obtain explanations for the credulous non acceptance of some argument for a given semantics (except the grounded semantics). Their work is very similar to that of [SWW20] presented previously. The differences here are that the authors consider both induced and spanning subgraphs for their explanations. More importantly, the subgraphs are not the explanations themselves, but rather used to characterize explanations. To be more precise, they call a set of arguments (respectively of attacks) an explanation if the induced subgraph (respectively spanning subgraph) computed using this set does not credulously accept the queried argument, and nor does any of its supergraphs.

[UW21] proposes strong explanations for credulous acceptance of a set of arguments under a given semantics. A strong explanation is a set of arguments such that for every subgraph induced by a superset of the explanation, there exists an extension of the considered semantics that includes the set to explain. Thus, strong explanations for credulous acceptance can be seen as a core set of arguments needed for an argument to be part of at least one extension under the desired semantics. Here again, subgraphs are not the explanations themselves, but are used as a tool to define explanations.

A specific kind of graph that is also used in explaining argumentative results is *Defence Trees*. Defence Trees (sometimes also called *Dispute Trees*) are trees where nodes are arguments and each successor of a node is an attacker of that node. As such, they can be used to prove whether an argument is acceptable or not. Some works, like [RT21], use Defence Trees as explanations for argumentative results, while others, like [FT15a], use them to compute their notion of explanation. In [FT15a], the authors aim to explain the credulous acceptance of some argument under admissibility, and do so by defining a new semantics, *related admissibility*, which is used to characterize explanations. Dispute Trees (more precisely, *Dispute Forests*) are used to compute these explanations. In [RT21], the authors argue that a Defence Tree is a dialogical explanation for the selection of an argument in an extension since it can be used to show that it is acceptable. They motivate this choice by stating that Dispute Trees can be seen both as contrastive and selective explanations. Contrastive because Dispute Trees can be seen as a debate in which the person putting the initial argument forward must defend it against counter arguments. Selective because one could deem one branch instead of the entire tree as a sufficient explanation.

Changes. We now turn to the second category, which concerns *changes*. Changes consist in identifying which elements to remove from an Argumentation Framework in order to modify a given result. This kind of methods can be drawn back to the problem of dealing with inconsistencies, and in particular trying to restore consistency, in knowledge representation formalisms.

This is the method used in [FT15b], in which the authors explain why an argument is not credulously accepted under admissibility. Their explanations consist of sets of arguments or attacks to remove from the Argumentation Framework in order to make the considered argument credulously accepted under admissibility in the resulting subgraph. These explanations are computed using Defence Trees.

Although they were not considered through the prism of explainability, such sets were also studied in [UB19], in which they were called "*diagnosis*". The authors were concerned in cases of Argumentation Frameworks in which no arguments are credulously or skeptically accepted under some semantics, and how such frameworks could be "*repaired*". They investigated some fundamental problems regarding diagnoses and repairs, such as deciding if some exists, their verification, their computation and the enumeration of all existing solutions.

Diagnoses are also parts of the study of [NJ20], which we already discussed before in the category of explanations using subgraphs. In addition to their notion of explanations, the authors study the notion of diagnosis as another way of identifying underlying reasons for the non acceptance of an argument under credulous reasoning. In particular, they provide a way of computing both their explanations and diagnoses using logical formulas, and providing complexity results.

Extensions. The third category of approach consists of taking *sets of arguments* as explanations. This is probably the most widely used approach to this problem. In most of the works using this method, the point of view is to consider that explanation equates to justification and that arguing for an argument is justifying it, and thus explaining it. Hence the use of sets of arguments as explanations, since such sets can be deemed

as arguing for a queried argument.

In [FT15a], as we have already discussed before, the authors define an explanation semantics, called *related admissibility*, which provides all the reasons why an argument belongs to an admissible set. Even though Defence Trees are used to compute the extensions of this new semantics, it is the extensions which are deemed to be explanations. The idea of related admissibility is to get rid of all the arguments that are not relevant for the acceptance of the queried argument, that is, those that are not connected to it via the attack relation.

In [BB20b, BB21c], the authors propose a basic framework to compute explanations as sets of arguments for the credulous/skeptical acceptance or non-acceptance of an argument. They distinguish between skeptical and credulous explanations. Skeptical explanations provide all the reasons why an argument can be accepted and one reason why an argument cannot be accepted. On the other hand, credulous explanations provide one reason why an argument can be accepted and all the reasons why an argument cannot be accepted. Their framework for explanations can be parameterised in order to modify the way explanations are computed, for instance taking into account the *depth* to consider when computing an explanation. In their work, the authors focus on some human biases used to select explanations such as simplicity (taken as minimality), sufficiency and necessity. In subsequent works ([BB21a, BB21b]) the authors extend their framework to adapt it to Structured Argumentation (adding another parameter to control the form of the explanation) and to compute contrastive explanations (the intersection of why an argument (the fact) is accepted and why a set of arguments (the foils) are not). As such, a contrastive explanation contains the reasons for both the acceptance (or non acceptance) of the fact and the non acceptance (or acceptance) of every foil. The authors also provide a way to deal with implicit foils both for Abstract Argumentation and Structured Argumentation.

Some other works define their explanations from the observation that in the computation of an extension, some parts are non-deterministic choices, while others, deterministic, result from the first ones. This is the case of [LvdT20], in which the authors base their approach on the observation that each Strongly Connected Component (SCC) of an Argumentation Framework can be seen as making a choice for accepting conflict-free sets of arguments. From these choices results the rest of the accepted arguments. Thus, in a set of arguments, each argument can be explained by the set of arguments that were chosen in a given SCC.

Similarly, in [BU21], the authors observe that complete and admissible semantics are computed firstly by the computation of the grounded (respectively strongly admissible) extension, then making choices in even cycles, and finally computing the grounded (respectively strongly admissible) extension again. As such, they define the arguments chosen in the even cycles as the explanations for some complete or admissible extension.

Dialogues. The fourth, and last, category regroups approaches that use so-called dialogues (or dialogue-games) as explanation. Dialogues are a formalism which allow agents (usually two, but it can be more) to engage in a conversation. This conversation is built turn by turn by the agents which can use *moves* to put forward elements in the discussion. The formal part comes as a *protocol* that the agents must follow in order to use their moves. This allows to use dialogues as proofs of certain results (typically, in a given situation, the dialogue will necessary reach a given state). Such proofs are called *dialectical proofs*.

[MC09] studies how dialogues (called argument games) can be used to prove certain results in argumentation. These dialogues are structured around two agents, the proponent and the opponent. An initial argument is put forward by the proponent as a first move and then the two agents alternate in attacking each other's arguments. The initial argument can be proved to have a certain status if the proponent has a winning strategy for the dialogue. The nature of this certain status is dependent on the protocol used in the dialogue. It should be noted that this particular work is not necessarily tied with explainability, but provides good insights on the workings of dialogues and how to use them.

The authors of [BGK$^+$14] take changes as explanations for argumentative results, but use dialogues to obtain them. The dialogues are structured as detailed before, with a proponent and an opponent attacking each other's arguments alternatively. The idea is for the proponent to consider several Argumentation Frameworks at a time, close to each other but different by the means of changes. Then, the proponent can

make moves that are legal in any the AFs considered, but at each move, the AFs in which it is illegal are removed. The AFs that are still present at the end of the dialogue (so those in which all the moves of the proponent are legal) are deemed to be explanations for the argumentative result the dialogue is about (again, depending on the protocol used).

[ABC17] proposes a dialogue with explanatory capabilities using Abstract Argumentation in the context of inconsistent databases. The main concern of the authors is to have a way to solve inconsistencies in a database without having to use classical repairing techniques, which involve the removal of information. Instead, they want a way to yield results even in the presence of inconsistencies, hence their choice of using Abstract Argumentation. In this setting, the dialogue is taken to be the explanation that answers to the user's query.

In [SA18], the authors present their argumentation-based dialogue framework for explanation in a human-robot interaction setting. The human and the robot are supposed to cooperate in a Treasure Hunt Game. The authors model the beliefs of both agents, and their dialogues, which result in Argumentation Frameworks, are designed so that the agents can mutually change their beliefs, either by adding new ones or modifying the existing ones. As such, the dialogues produced have an explanatory role. In particular, the authors try to measure how the dialogues help to achieve the task at hand, and if certain kinds of dialogue, tailored towards specific goals, were more efficient in helping than others.

We now present some additional aspects of explanations. Contrarily to the works mentioned previously, these are not specifically related to Abstract Argumentation, but to Explainability in general. As such, they should be taken as high level ideas on explanations, the different types that there can be, how to obtain them, what can they be used for, etc... By no means this short presentation should be considered exhaustive, each of ideas mentioned could be (and certainly are) the subjects of entire works on their own.

Abductive explanations. Abduction is form of logical reasoning, like deduction and induction, which was formulated by the American philosopher Charles Sander Peirce during the 19th century. The idea of abduction is to find what led to some event that is observed. This can be formally represented with the following logical scheme: if we know b and that a leads to b, then we can conclude a. It should be noted that abduction, unlike deduction, does not yield a conclusion that we can be sure of, only a plausible one. Nonetheless, Peirce considered abduction as the only form of reasoning that could introduce new knowledge.

More recently, abduction has been studied as a mean to obtain explanations. Indeed, we can see from its logical scheme that abduction can be used to infer what can be considered as causes of an observed event. For this reason, among the other names that can be used to designate abduction, it has been called "inference to the best explanation" ([Har65]). The process is to formulate hypotheses on what caused a certain observed event to occur, and to identify among these hypotheses which one can be considered as the "best". Although it is usually a difficult task to automate, humans are generally considered quite efficient to do it. The usual take is that humans use biases, like simplicity, temporal closeness, etc... to efficiently select what they deem the best hypothesis as the explanation (see [Mil19]). Still, the explanations obtained this way are thus called *abductive explanations*.

Counterfactual explanations. Counterfactual thinking is a concept of psychology ([Roe97]). It designates the process of creating possible alternative to events that have already occurred. In a nutshell, it can be understood as corresponding to the thought process induced by the questions "What if..." or "If only...". As its name states, it goes "contrarily to the facts". Humans are quite adept at imagining how reality could have turned out, provided that things went differently. And here we touch the core of what constitutes counterfactual thinking and how it works. It is important to understand that it is based on *differences*, or *changes*, that humans make to generate new situations that are different from the one they experienced. Because with this understanding, we can move from counterfactual thinking to counterfactual explanations.

Indeed, counterfactual thinking can be used to seek and obtain explanations. Consider that explanations are usually sought for in response to a surprising situation. It is generally accepted that surprise comes from making predictions as to what is supposed to happen and then observe something else. Thus, to make

their prediction models more accurate, humans will naturally try to explain why the situation they observed turned out *different* from the situation they expected. This is where counterfactual thinking comes into use. By using counterfactual thinking, and putting alternative possibilities in contrast, humans are able to pinpoint a selection of causes that, if changed, would have led to the situation they expected in the first place. These causes indicate what should be changed in the prediction model used to make the first expectation, and are usually considered to be the (counterfactual) explanation. Additionally, it should be noted that counterfactual explanations are often rather simple. This is because the process of counterfactual thinking relies on changes made on an already existing situation. To generate alternatives, a small number of changes suffices, as all the rest can be deemed to be the same as in the initial situation. Thus, the generation process is kept efficient, and the explanations (so the changes made) are kept as small as possible.

This concludes our presentation of related works on explanations for Abstract Argumentation and explanations in general. There exist other types of explanations we could have mentioned, but we consider them to not be relevant to the present work. As we have seen, explanations related to Abstract Argumentation can have many forms, even for explaining the same result. This gives them a good potential for adaptability to the preferences of the person asking for them. We now move on to motivate our own work and to present our main hypotheses.

3.2 Motivation and Hypotheses

3.2.1 Motivation

To introduce our motivation for our approach, let us present an example from [BB20a], slightly reduced and adapted. This is a real-case scenario of Abstract Argumentation applied for the Dutch National Police.

Example. A citizen has ordered a product through an online shop, paid for it, and received a package. However, it is the wrong product, it seems suspicious as if it might be a replica, rather than a real product. Still, the citizen wants to file a complaint of internet trade fraud. While the citizen provides the information from the described scenario, the system constructs further arguments from this, based on the Dutch law. The following arguments are obtained (their conclusions are emphasized):

A_1 It is not because the wrong product was received, that it is a case of fraud; then we may consider that it is *not a case of fraud*.

A_2 It is not because the wrong product was received, that the counterparty has not delivered; the *counterparty has delivered*.

A_3 A suspicious product is usually fake, which supports the fact that the *product is fake*.

A_4 The reasons which lead to the conclusion that the product is fake, and the fact that when a product is fake, then usually the counterparty did not deliver, lead to the conclusion that the *counterparty did not deliver*.

A_5 An investigation shows that there is no problem with the product: the *product is not fake*.

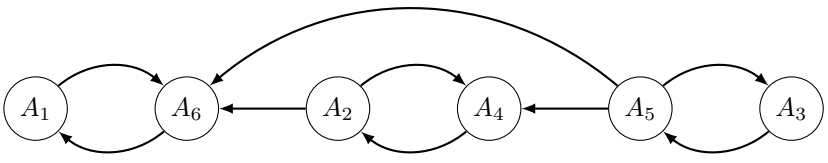

Figure 3.1: Delivery Example from [BB20a]

A_6 The fact that the complainant paid and was delivered, combined to the assumption that the product is fake and to the other reasons which lead to the conclusion that the counterparty did not deliver, shows that it is likely to be a *case of fraud*.

The arguments of this scenario and their attack relationships can be represented by the Argumentation Framework depicted on Figure 3.1.

Now, the question is: which conclusion can be drawn from this situation? Of course, we want to know whether this is a case of fraud or not. A_1 concludes that it is not a case of fraud, whereas A_6 concludes that it is a case of fraud. We may naturally want to look for extensions, that will then act as our *decision*, that contain these arguments. Let us pick stable semantics to take our decision. The stable extensions are: $\{A_6, A_4, A_3\}$, $\{A_1, A_2, A_5\}$, $\{A_1, A_4, A_3\}$. Since we have several choices, we may want use the techniques of credulous or skeptical acceptance presented in Section 2.2 to support a preference for either A_1 or A_6. In this case, both are credulously accepted and none is skeptically accepted (still under stable semantics). Thus, we do not really have elements to choose one over the other.

It may result from this that Abstract Argumentation alone (or at least, the rudimental techniques we have used) may not lead us directly to a decision. Most certainly, deciding whether this is a case of fraud or not will either rely on more advanced techniques, or on some external factors outside of the argumentative process. Nonetheless, we may require explanations as for why Abstract Argumentation cannot help us much in this situation. In particular, we may want to ask why are A_1 and A_6 both credulously accepted under stable semantics, and why are they both not accepted under stable semantics. This is indeed the questions that are tackled in the related works presented in the previous section.

However, we wish to *strongly insist* on the fact that this makes sense because of *the decision we want to make* and because of *the meaning of the arguments*. Indeed, we knew *from the start* that we were to decide whether the scenario was a case of fraud or not. And we have, in the Argumentation Framework, arguments *whose meaning is precisely the decision we want to make*. As such, the Argumentation Framework is, in fact, *built around those arguments*. They are at the core of the framework, and the other arguments either attack one of them and support the other, or attack the latter and support the former. This is why questions of credulous / skeptical acceptance are relevant here, because of our decision relying on the acceptability status of some precise arguments. But, *this might not be the case for all Argumentation Frameworks*.

We should not forget that Abstract Argumentation provides tools to select arguments *collectively*. Thus, focusing on some precise arguments may not always be relevant. Consider this: for what other reason should an argument be individually focused on, other than its *internal meaning*? And recall that internal meanings of the arguments *are left aside in an abstract setting*.

As such, in the present work, we propose to tackle questions relative to the basic utility of Argumentation Frameworks: the selection of semantics extension. To come back to our previous example, suppose that in the end, the extension $\{A_1, A_2, A_5\}$ is chosen. A user that wishes for explanations may then ask `"Why is this a valid result?"`. Since $\{A_1, A_2, A_5\}$ was chosen on the basis of being a stable semantics, this in fact equates to wondering why $\{A_1, A_2, A_5\}$ is a stable extension. More generally, given a set S of arguments and a semantics σ, we are interested in the following questions:

Q_σ^{Ext} : `"Why is` S `[not] a` σ `extension?"`

Note. Please note that since these questions are referring to the basic process of argumentative selection, they are relevant in any context involving Abstract Argumentation.

Notice that question Q_σ^{Ext} is just a modification of the question `"Is` S `[not] a` σ `extension?"`, which just asks to solve the Verification Problem mentioned in Section 2.2. As such, we will consider question Q_σ^{Ext} to ask to solve a modified version of the Verification Problem, which we call the *Explainable Verification Problem* and denote $XVer_\sigma$, and that we define as the following:

$XVer_\sigma$ **Input :** an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a semantics σ, a set of arguments $S \subseteq \mathscr{A}$
Output : the reasons that make S an extension (or not) of σ in $\mathcal{A}$

3.2.2 Hypotheses

In order to provide answers for questions Q_σ^{Ext}, we rely on some hypotheses that we make clear in this section. First of all, it is commonly agreed that, when dealing with finding and giving explanations, *contextual information* is critically important. Indeed, the need for explanation rises *in reaction* to some event, usually an *unexpected* one. And an event is deemed unexpected when it does not make sense with the context in which it occurred. Hence the importance of contextual information. In our case, we describe the context of our work using the following Hypothesis:

> *A user asks for an explanation after they have been presented the result of an Abstract Argumentation selection process by some program.* (H1)

In other terms, we place ourselves in a situation where some user needs to make a decision with an Abstract Argumentation tool. The decision of the argumentative process may be the final decision of the user, or only a step in chain of processes leading to the final decision, it does not matter. What matters is that the user *reacts to the decision of the argumentative process*. This result may typically be the selection of a set of arguments, that is to say *the computation of an extension of some semantics*. In this situation, the semantics used would correspond to the constraints under which the user wanted their decision (or decision step) to be made.

Vocabulary. In the following, we will refer to the program that has computed the result as *the system*.

From this, we identify that our context is made of three contextual information: the result that was presented by the system (a set of arguments), the way this result was obtained (a semantics), and the object from which it was obtained (an Argumentation Framework).

Notation. Except specified otherwise, we will denote them using the following notation:

- $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ denotes the contextual Argumentation Framework
- σ denotes the contextual semantics
- $Res \subseteq \mathscr{A}$ denotes the result given in the context

Example. Consider $S \subseteq \mathscr{A}$ and the question Q_σ^{Ext}: `"Why is` S `[not] a` σ `extension?"`. σ then refers to the semantics used to compute Res in the contextual Argumentation Framework $\mathcal{A}$. Both $\mathcal{A}$ and Res are not referred to in the question, hence their reference is left implicit. Note that we do not necessarily have $S = Res$.

Our next hypothesis concerns the explanatory process. We have already stated that the need for explanation rises *in reaction* to some event (here the computation of an extension). In our situation, this reaction takes the form of a *question* asked by the user. It is through the question that they ask that the user requires an explanation. Hence, the explanation we will provide will in turn be in reaction to the question of the user. In other terms, our explanations are in fact answers to the questions of the user.

> *An explanation is an answer to some question.* (H2)

Note that, in the present work, we consider that the user only asks why-questions. *This does not mean that explanations are necessarily answers to why-questions in general.* A question like `"How was` Res `obtained?"` may be deemed as another question asking for explanations. Chances are that this explanation may even be different from the ones that we will seek to provide in answer to Q_σ^{Ext}, although the two questions seem very similar in the case where $S = Res$. Note also that Hypothesis (H2) does not imply that *every question asks for an explanation*. The following results directly from Hypothesis (H2).

Corollary 1. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and σ be an Abstract Argumentation semantics. An explanation of S for σ on $\mathcal{A}$ is an answer to Q_σ^{Ext}.*

As such, in the following, to define explanations, we will in fact aim at defining answers to Q_σ^{Ext}.

Finally, our last hypothesis will, in practice, constrain how our explanations are defined. Indeed, our work is done in the perspective that Abstract Argumentation will become widely deployed as a decision making tool, and may thus come to be used by anybody. In particular, we consider that even users without any idea of Abstract Argumentation selection processes operate may come to use it.

The user has no expert knowledge of Abstract Argumentation. (H3)

Observe that even a user that does not know how the system works may require explanations about it. Hence, our explanations should take this into account, and should be understandable and usable by anybody.

3.3 Technical Tool: Graph Theory

Before to define our explanations, we recall here some elementary notions of Graph Theory that we will use in this chapter. We refer the reader to [BM08] for additional notions on this subject.

We suppose the reader familiar with the notion of graph itself, so we begin with the notion of subgraph. A subgraph of some graph is basically another graph, included in the first one.

Definition 22. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. G' is a *subgraph* of G iff $V' \subseteq V$ and $E' \subseteq E$.

Note. In Definition 22, G is called a *supergraph* of G'.

Note. Note that an arbitrary graph is always a subgraph of itself. It is possible to have a more strict notion in the case that is not wanted.

Definition 23. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. G' is a *strict subgraph* of G iff it is a subgraph of G and either $V' \subset V$ or $E' \subset E$.

Note. In Definition 23, G is called a *strict supergraph* of G'.

A given graph may potentially have a very large number of subgraphs. It may thus be of interest to consider only specific ones. That is the case for induced and spanning subgraphs.

Definition 24. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. G' is an *induced subgraph of G by V'* if G' is a subgraph of G and for all $a, b \in V', (a, b) \in E'$ iff $(a, b) \in E$. G' is denoted as $G[V']_{Ind}$.

Definition 25. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. G' is a *spanning subgraph of G by E'* if G' is a subgraph of G and $V' = V$. G' is denoted as $G[E']_{Span}$.

As such, induced subgraphs are those for which we keep only a certain number of the original nodes, as well as all the arcs that are related to these nodes. Spanning subgraphs are those for which we keep only a certain number of arcs.

Example. Figure 3.2 depicts an induced subgraph of Figure 2.2 by the set $\{a, b, c, d, e\}$. Figure 3.3 depicts an induced subgraph of Figure 2.3 by the set $\{a, b, c, d, g\}$. Figure 3.4 depicts a spanning subgraph of Figure 2.2 by the set $\{(f, e), (f, g), (h, e), (i, h), (i, j), (j, i)\}$. Figure 3.5 depicts a spanning subgraph of Figure 2.3 by the set $\{(c, f), (f, b), (f, d), (f, e), (f, h), (j, f)\}$.

Induced and spanning subgraphs are examples of ways to obtain a new graph from another single graph. Another interesting operation producing a new graph from other ones is the union that represents the aggregation of the information contained in two graphs.

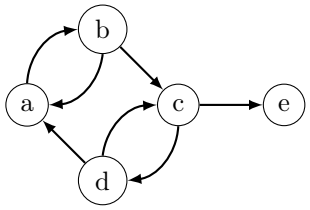

Figure 3.2: An induced subgraph of Figure 2.2

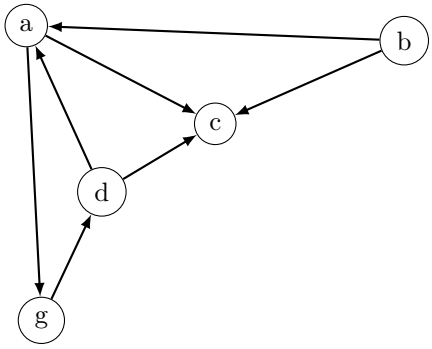

Figure 3.3: An induced subgraph of Figure 2.3

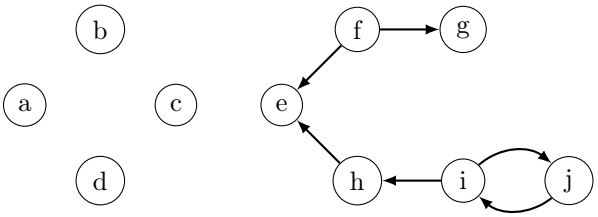

Figure 3.4: A spanning subgraph of Figure 2.2

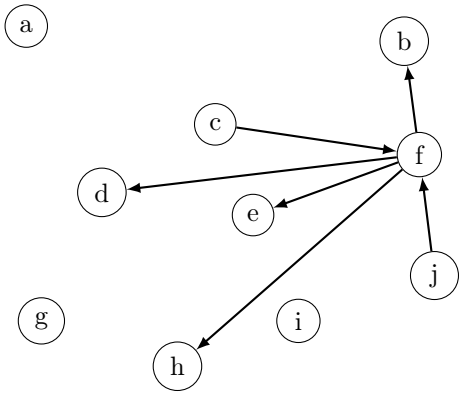

Figure 3.5: A spanning subgraph of Figure 2.3

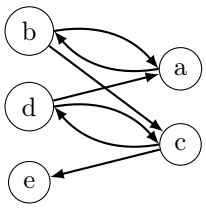

Figure 3.6: The graph of Figure 3.2 rearranged to highlight its bipartite nature

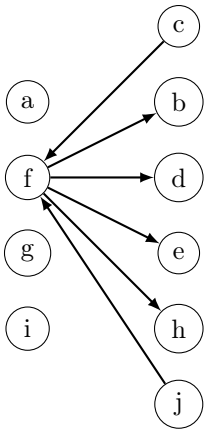

Figure 3.7: The graph of Figure 3.5 rearranged to highlight its bipartite nature

Definition 26. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs. The *union* of G_1 and G_2 is $G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2)$.

Example. Let G_1 be the graph of Figure 2.2, G_2 be the graph of Figure 3.2 and G_3 be the graph of Figure 3.4. We obviously have $G_1 = G_2 \cup G_3$.

We now consider a particular kind of graphs, bipartite graphs. Bipartite graphs are those graphs whose set of vertices can be split two disjoint sets and in which every arc connects a vertex of one part to a vertex of the other part:

Definition 27. Let $G = (V, E)$ be a graph. G is *bipartite* (with *parts* T and U) if and only if there exist $T, U \subseteq V$ such that $T \cup U = V$ and $T \cap U = \varnothing$ (T and U are a partition of V) and for every $(a, b) \in E$, either $a \in T$ and $b \in U$, or $a \in U$ and $b \in T$.

Note. A bipartite graph (V, E) with parts T and U can be denoted (T, U, E).

Note. For a bipartite graph (T, U, E), we say that U is the *complement part of* T and vice-versa.

Example. Figures 3.6 and 3.7 depict the graphs of Figures 3.2 and 3.5 respectively, but visually rearranged so that their bipartite nature is more obvious. Here, the graph of Figure 3.2 is shown with parts $\{b, d, e\}$ and $\{a, c\}$, while the graph of Figure 3.5 is shown with parts $\{a, f, g, i\}$ and $\{b, c, d, e, h, j\}$.

Note. Notice that a given bipartite graph may have its vertices separated in several possible ways. For instance, the graph of Figure 3.7 could have been shown with parts $\{f\}$ and $\{a, b, c, d, e, g, h, i, j\}$.

We now define the successor and predecessor functions of a graph, which allow to have a sense of neighborhood in that graph.

Definition 28. Let $G = (V, E)$ be a graph. The *successor* function of G is the function $E^+ : V \mapsto 2^V$ such that $E^+(v) = \{u \mid (v, u) \in E\}$ and the *predecessor* function of G is the function $E^- : V \mapsto 2^V$ such that $E^-(v) = \{u \mid (u, v) \in E\}$.

Note. The successor and predecessor functions can be extended to sets of vertices.

Definition 29. Let $G = (V, E)$ be a graph and S be a set of vertices. $E^+(S) = \bigcup_{v \in S} E^+(v)$ and $E^-(S) = \bigcup_{v \in S} E^-(v)$.

Finally, since the successor and predecessor functions capture a certain notion of neighborhood, we may want to parameterize this notion. For instance, we might want to consider that the neighbors of some vertex are not only those that are directly connected to it via an arc, but also those that are connected to them. In that respect, we introduce a n-step version of the successor and predecessor functions.

Definition 30. Let $G = (V, E)$ be a graph and $n \geq 0$. The *n-step* successor (resp. predecessor) function of G is $E^{+n}(v) = \overbrace{E^+ \circ \cdots \circ E^+}^{\text{n times}}(v)$ (resp. $E^{-n}(v) = \overbrace{E^- \circ \cdots \circ E^-}^{\text{n times}}(v)$).

Convention. By convention, we have $E^{+0}(v) = E^{-0}(v) = v$.

Note. Note that $E^{+1}(v) = E^+(v)$ and $E^{-1}(v) = E^-(v)$.

Note. In the context of an Argumentation Framework, the successor (resp. predecessor) function represents the arguments that are attacked by (resp. the attackers of) some argument(s). An Argumentation Framework being usually denoted by (A, R), the successor and predecessor functions are thus denoted R^+ and R^- in this context.

Example. In the graph of Figure 3.3, we have $E^{+1}(a) = \{c, g\}$, $E^{+2}(a) = \{d\}$ and $E^{-1}(a) = \{b, d\}$. Additionally, $E^{+1}(\{b, g\}) = \{a, c, d\}$.

Finally, we consider some vertices having a particular status in a graph.

Definition 31. Let $G = (V, E)$ be a graph and $v \in V$. v is a *source vertex* if and only if $E^-(v) = \varnothing$.

Definition 32. Let $G = (V, E)$ be a graph and $v \in V$. v is a *sink vertex* if and only if $E^+(v) = \varnothing$.

Definition 33. Let $G = (V, E)$ be a graph and $v \in V$. v is an *isolated vertex* if and only if it is both a source vertex and a sink vertex.

Thus, source vertices are vertices for which there exists no arc in the graph targeting them. Sink vertices are those for which there exists no arc in the graph originating from them. Isolated vertices are those completely devoid of connections to the other vertices.

Example. In the graph of Figure 3.7, c and j are source vertices, b, d, e and h are sink vertices, and a, g and i are isolated vertices.

3.4 Visual Explanations for Argumentation Semantics

We propose, here, to provide formal answers to a certain kind of questions. These questions are those introduced in Section 3.2.1: `"Why is S [not] a` σ `extension?"`, S being a set of arguments and σ ranging over the conflict-free, admissible, complete and stable semantics.

Example. Consider the Argumentation Framework depicted on Figure 2.2. Imagine that a user asks the question `"Why is {b,d} a complete extension?"`. In this case, $\{b, d\}$ is a complete extension (see Table 2.2), thus we would need to provide the elements that *show* what makes $\{b, d\}$ to be so. Alternatively, imagine that a user asks the question `"Why is {a,b} not a complete extension?"`. In this case, $\{a, b\}$ is not a complete extension (again, see Table 2.2), hence we should provide the elements that *show* the reasons for this set not to be so.

The next step is to discuss what these elements are. To answer those questions, and provided that the user can access to the Argumentation Framework that is being used, notice that they could simply check the conformity of the questioned set with Definition 4. However, not only may the user not have direct access to the Argumentation Framework, but it also requires expert knowledge (in particular, of the definitions) and might prove to be tedious, because the graph might be large and/or contain a large number of arcs (think of the Argumentation Framework of Figure 2.3 for instance). On the other hand, it is worth pointing out that the conformity to this definition is *precisely* what makes a set of argument valid or not regarding a given semantics, and thus contains *all the reasons* to decide on its validity status. As such, what we propose are ways to do this conformity check, such that:

1. The conformity check should be as easy as possible
2. The user should not get lost in information

3. No expert knowledge is required

To achieve the objectives we have just listed, we take advantage of the visual nature of Argumentation Frameworks. Argumentation Frameworks are graphs. Graphs can be *drawn*. Even if this property is difficult to mathematically formalize, and so to reason on, we believe it is *decisive* in how it can make explanations understandable and usable by anyone. Thus, our explanations will be graphs as well.

In compliance with Point 2, we will aim for our explanations to keep only the information that might be useful[1] for the user. To do so, we reverse the problem, and in fact get rid of all the information that we can be sure is irrelevant. As such, our explanation graphs will be parts of the initial information from which the decision was taken. In more formal terms, our explanations will be subgraphs of the initial Argumentation Framework.

To achieve Point 3, we will aim for our conformity check to rely only on *visual* properties of our explanations. The idea is that this way, users can make the conformity check themselves, using only the *layout* of the explanation, thus getting past the need for expert knowledge of the precise formal definitions. We will still support such a use of our explanations via formal results. So, we need a formal counterpart to this notion of "visual property". Formally, we identify them with *structural* properties of the explanation graphs.

Finally, to respect Point 1, we will also take advantage of the decomposition of Abstract Argumentation semantics into principles presented in Section 2.4. Indeed, semantics can be decomposed into *simpler* and *modular* underlying principles. As such, checking conformity with a given semantics amounts to checking conformity with each of its underlying principles. Obviously, this means that, for one given semantics, the user could possibly do several conformity checks. However, they will all individually be simpler than if the user was to check conformity with the entire semantics at once. Additionally, since some principles are underlying in several semantics, there won't be a lot of different explanations to define. Moreover, we are confident that through repeated uses, the principles that are underlying in several semantics will get easier (and so, faster) to check, thus mitigating the drawback of possibly having to do several checks for one semantics. So, our explanations will in fact be explanations for semantics principles, instead of explanations for semantics. With this approach, an explanation for a semantics will then be the set of explanations for its underlying principles, the latter being in turn the answers to the following intermediate questions where π represents an Abstract Argumentation principle:

Q_{π}^{Ext}: `"Why does S [not] respect the` π `principle?"`

To summarize, in this work, explanations are subgraphs of an initial Argumentation Framework and are meant to be used to check conformity with the underlying principles of semantics using only structural properties. Explanations for semantics are then just the set of explanations for the underlying principles that constitute the given semantics.

3.4.1 Methodology

In the following subsections, we formally present our explanations. Each time, we follow the same methodology:

1. We begin with an introductory example which we discuss to identify what the explanation is about and what is required;
2. We then formally define our explanation (possibly in several steps);
3. Once formally defined, we illustrate the explanation with further examples and introduce notations.

To support the use of our explanations, we will investigate some of their properties in Section 3.4.6. Additionally, we end Section 3.4 with a wider example, aimed at illustrating how we envision the entire explanatory process to take place.

Finally, we will use the following color scheme on all the figures representing our explanations:

[1] We say "might be useful" only, because we consider that we don't know what is relevant or not for the user in advance.

Legend of the colors used in the explanation graphs:

- The arguments of the set which is given as input in the question will be in *blue*.
- Arguments that are displayed because they take the role of attackers (of a given set), as well as problematic attacks, will be in *red*.
- Arguments that are defended or acceptable with respect to the input set, but not part of it, will be in *green*.
- Other arguments and attacks of the explanation subgraph will appear in *black*.
- Elements of the explanation graph that make a conformity check fail will be displayed as **thickened**

3.4.2 Explanation for Coherence

Recall that a set of arguments respects the Coherence principle if and only if there are no arcs between its elements. Hence, if we are to show why a set of arguments is coherent, we must show a part of the graph that highlights the absence of arcs within this set. We begin with an example in order to provide an intuition of how to define the explanation.

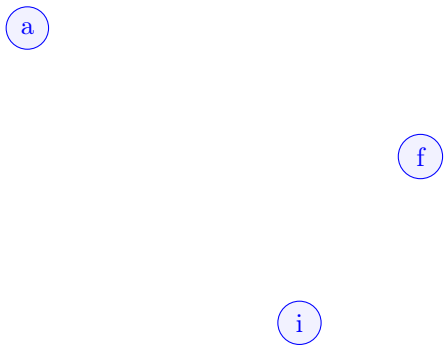

Figure 3.8: Explanation on why $\{a, f, i\}$ is coherent in Figure 2.3. All arcs between a, f and i are shown, and there is none.

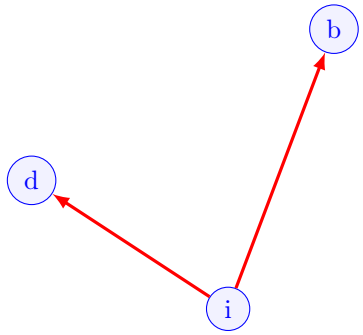

Figure 3.9: Explanation on why $\{b, d, i\}$ is not coherent in Figure 2.3. All arcs between b, d and i are shown, and there is one between i and b, and one between i and d.

Example. Consider the Argumentation Framework of Figure 2.3 and the questions `"Why does {a,f,i} respect the Coherence principle?"` and `"Why does {b,d,i} not respect the Coherence principle?"`. Figures 3.8 and 3.9 show the answers for the first and second question respectively.

The idea here is to have a subgraph in which we make sure that if there is at least one arc between two arguments of the questioned set, then such an arc appears in it. Indeed, the presence of at least one such arc is enough to conclude that the set is not coherent. If no such arc exists, then none is displayed and we can conclude that the set is coherent.

Note. In this situation, we may only focus on the arguments of the questioned set, and forgo all the others.

Note. Even if only one arc between two arguments of the set is enough, there may very well be several of them (for instance, on Figure 3.9). In this case, we may choose to keep only one, all of them, or even any number inbetween. All these choices result in valid explanations, since we can infer from them that the set is not coherent. It is however crucial that at least one appears so that we do not make a false conclusion.

All these considerations give rise to the following definition.

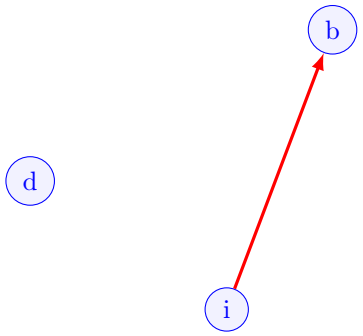

Figure 3.10: Other possible explanation on why $\{b, d, i\}$ is not coherent in Figure 2.3.

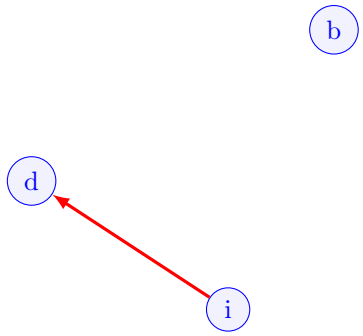

Figure 3.11: Other possible explanation on why $\{b, d, i\}$ is not coherent in Figure 2.3.

Definition 34. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, and consider $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Coh} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = S$
- $\mathscr{R}' \subseteq X$
- If $X \neq \varnothing$, then $\mathscr{R}' \neq \varnothing$

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$ by $Expl_{Coh}(S)$.

Example. We have already seen that Figure 3.9 shows an answer to Q^{Ext}_{Coh} for $\{b, d, i\}$ on the Argumentation Framework of Figure 2.3. Figures 3.10 and 3.11 depict other possible answers for the same question.

Once an answer to Q^{Ext}_{Coh} is provided, the conformity check simply consists in verifying whether or not an arc is displayed.[2] In other terms, it consists in verifying whether or not the set of arcs is empty.

Notation. In the following, considering a set of arguments S and an explanation for Coherence for S $Expl_{Coh}(S)$, we denote by C_{Coh} the condition "there exists no arcs in $Expl_{Coh}(S)$".

Explanation for Conflict-freeness

Recall that the conflict-freeness semantics is only made of the Coherence principle. As such, we can simply define an *answer to* Q^{Ext}_{CF} for a set S as being a set containing only $Expl_{Coh}(S)$.

Definition 35. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq \mathscr{A}$. An *answer to* Q^{Ext}_{CF} for S on $\mathcal{A}$ is a set $\{Expl_{Coh}(S)\}$.

Example. Figures 3.8 and 3.9 both display answers to Q^{Ext}_{CF} on the Argumentation Framework of Figure 2.3, for $\{a, f, i\}$ and $\{b, d, i\}$ respectively.

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{CF} for S on $\mathcal{A}$ by $Expl_{CF}(S)$.

3.4.3 Explanation for Defence

Recall that a set of arguments respects the Defence principle if and only if all its arguments are acceptable with respect to it. In particular, this means that for every argument that attacks one in the set, there is an

[2] Note that this is an inherently structural, and so visual property

argument in the set that attacks this attacker. Therefore, if we are to show why a set of arguments only contains arguments that are acceptable with respect to it, we must exhibit a part of the graph highlighting that all the attackers of this set are attacked in return. We begin by illustrating on some examples to give the intuition, and then formally define these explanations.

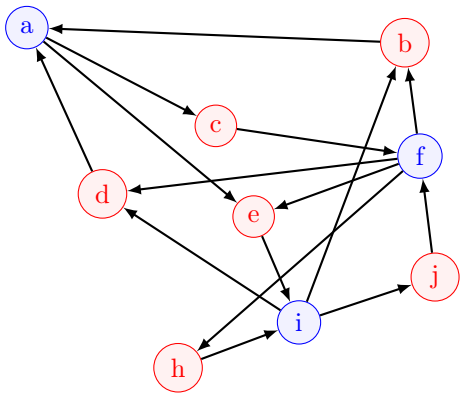

Figure 3.12: Explanation on why $\{a, f, i\}$ only contain arguments that are acceptable with respect to $\{a, f, i\}$ in Figure 2.3. All the attackers of a, i, and f are attacked in return.

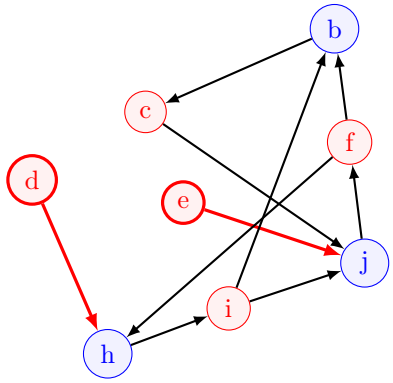

Figure 3.13: Explanation on why $\{b, h, j\}$ does not only contain arguments that are acceptable with respect to $\{b, h, j\}$ in Figure 2.3. h is attacked by d and j is attacked by e, while both d and e are not attacked by b, h or j in return.

Example. Consider the Argumentation Framework of Figure 2.3 and the questions `"Why does {a,f,i} respect the Defence principle?"` and `"Why does {b,h,j} not respect the Defence principle?"`. Figures 3.12 and 3.13 show the answers for the first and second questions respectively.

The idea here is to have a subgraph in which we make sure that, for every attacker of the set, if there is at least one arc from an argument of the set to that attacker, then such an arc appears in the subgraph. Indeed, the presence of such an arc is enough to conclude that the attacker we consider is being taken care of. Hence the need to do so for every attacker. If no such arc exists, then none is displayed and we can conclude that there is a hole in the defence of the set, and so that it does not respect the Defence principle. Additionally, the subgraph should contain the arcs from the attackers to the set, to show that they are indeed attackers.

Note. In this situation, we may only focus on the arguments of the questioned set and its attackers, and so forgo all the others. Additionally, we are not interested in attacks between two attackers of S, or between two elements of S. So, we can only keep the arcs that go from an element of S to an attacker of S or vice versa.

Note. We can make a similar observation as in the case of explanations for the Coherence principle, concerning the arcs from the questioned set to its attackers. For each attacker, even if only one is enough, there may very well be several of them (for instance, on Figure 3.12, both f and i attack d). Just as in the case of explanations for the Coherence principle, we may in this situation choose how many we keep between one and all of them, all these choices resulting in valid explanations. Yet again, it is crucial that at least one appears, so that we do not make a false conclusion.

All these considerations give rise to the following definition.

Definition 36. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, and consider $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Def} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$
- $X \subseteq \mathscr{R}' \subseteq X \cup Y$

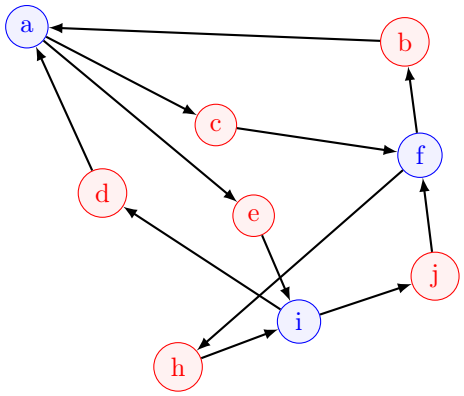

Figure 3.14: Other possible explanation on why $\{a, f, i\}$ only contains arguments that are acceptable with respect to $\{a, f, i\}$ in Figure 2.3.

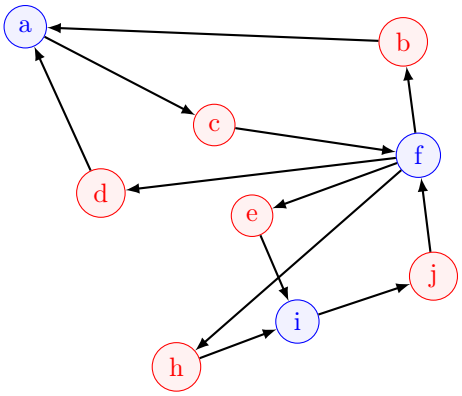

Figure 3.15: Other possible explanation on why $\{a, f, i\}$ only contains arguments that are acceptable with respect to $\{a, f, i\}$ in Figure 2.3.

- $\forall b \in \mathscr{R}^{-1}(S)$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a, b) \in \mathscr{R}'$ with $a \in S$

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$ by $Expl_{Def}(S)$.

Example. We have already seen that Figure 3.12 shows an answer to Q^{Ext}_{Def} for $\{a, f, i\}$ on the Argumentation Framework of Figure 2.3. Figures 3.14 and 3.15 depict other possible answers for the same question.

Once an answer to Q^{Ext}_{Def} is provided, the conformity check consists in verifying whether or not every attacker of S is attacked back by S. Since, by definition, in an answer to Q^{Ext}_{Def} attackers of S can only be attacked by elements of S, this reduces to verifying whether or not every attacker of S is attacked *at all*. In other terms, it consists in verifying whether or not every attacker of S is not a source vertex.[3]

Notation. In the following, considering a set of arguments S and an explanation for Defence for S $Expl_{Def}(S)$, we denote by C_{Def} the condition "there exists no source vertex among the attackers of S in $Expl_{Coh}(S)$".

Explanation for Admissibility

Recall that the admissibility semantics is made of the Coherence and the Defence principles. As such, we can define an *answer to* Q^{Ext}_{Adm} for a set S as being a set containing $Expl_{Coh}(S)$ and $Expl_{Def}(S)$.

Definition 37. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq \mathscr{A}$. An *answer to* Q^{Ext}_{Def} for S on $\mathcal{A}$ is a set $\{Expl_{Coh}(S), Expl_{Def}(S)\}$.

Example. Figures 3.8 and 3.12 constitute an answer to Q^{Ext}_{Def} on the Argumentation Framework of Figure 2.3, for $\{a, f, i\}$.

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Adm} for S on $\mathcal{A}$ by $Expl_{Adm}(S)$.

3.4.4 Explanation for Reinstatement

Recall that a set of arguments respects the Reinstatement principle if and only if all the arguments that are acceptable with respect to it belong to it. In particular, this means that all the arguments for which the set

[3]Note that the presence of a source vertex is indeed a structural property.

attacks all the attackers should belong to it. However, in virtue of Definition 2, this *also* means that all the arguments *for which there exists no attacker* should belong to the set as well.

Now, these two kinds of arguments are quite not similar by nature. The arguments for which there exists attackers that the set attacks are *connected* to it via the *binary relation* of the graph, and always in the same way: they are the arguments that belong to $\mathscr{R}^{+2}(S)$. On the contrary, the arguments for which no attacker exists may not be connected to the set, and even if they are, we may not have a unique way to reach them all from the set. As such, we have decided to treat these two cases separately.

In practice, this means that we have split the Reinstatement principle into two sub-principles: the one concerning the arguments for which there exists no attacker, and the one concerning the arguments for which the set attacks all the attackers. We call these two principles *Rein1* and *Rein2*, and for a set S, they have the following meaning:

Reinstatement 1 (*Rein1*):	All the unattacked arguments are in S
Reinstatement 2 (*Rein2*):	All the arguments for which S attacks all the attackers are in S

Of course, these two sub-principles can be considered as principles on their own. Thus we can use to consider concepts that are defined regarding Abstract Argumentation principles, for instance the questions Q^{Ext}_{Rein1} and Q^{Ext}_{Rein2}. As they make together the Reinstatement principle, following our methodology, we will define an answer to Q^{Ext}_{Rein} as a set containing an answer to Q^{Ext}_{Rein1} and an answer to Q^{Ext}_{Rein2}.

Figure 3.16: Explanation on why $\{f, h, j\}$ contains all the unattacked arguments in Figure 2.2. Only f is unattacked and it is part of the set.

Figure 3.17: Explanation on why $\{b\}$ does not contain all the unattacked arguments in Figure 2.2. Only f is unattacked and it is not part of the set.

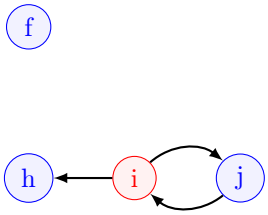

Figure 3.18: Explanation on why $\{f, h, j\}$ contains all the arguments for which it attacks all the attackers in Figure 2.2. f is unattacked and j attacks i, thus making h acceptable, and h is part of the set.

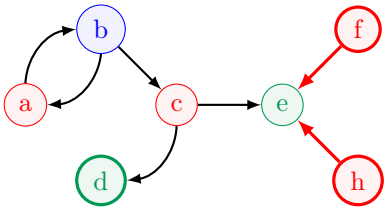

Figure 3.19: Explanation on why $\{b\}$ does not contain all the arguments for which it attacks all the attackers in Figure 2.2. b attacks c, thus defending e but it is not enough to make acceptable since it is also attacked by f and h, so e should not be in the set, which is the case. However, attacking c makes d acceptable, so d should be in the set, which is not the case.

Example. Consider the Argumentation Framework of Figure 2.2 and the questions `"Why does {f,h,j} respect the Reinstatement principle?"` and `"Why does {b} not respect the Reinstatement principle?"`. Figures 3.16 and 3.18 show the answer for the first question, while Figures 3.17 and 3.19 show the answer for the second question.

Explanation for *Rein1*

In the case of *Rein1*, the idea is to have a subgraph in which we make sure that, if an unattacked argument does not belong to the questioned set, then such an argument appears in the subgraph. Indeed, the presence of such an argument is enough to conclude that there exists an argument that is acceptable with respect to the set and does not belong to it.

Note. In this situation, we may only focus on the arguments that are not attacked, and forgo all the others. Additionally, since the arguments displayed are unattacked, there can't exist any interaction between them, thus we will always have an empty set of arcs.

Note. We can make a similar observation as in the previous cases, concerning the unattacked arguments that are not in the set. Even if only one is enough, there may very well be several of them. Just as in the previous cases, we may in this situation choose how many we keep between one and all of them, all these choices resulting in valid explanations. Yet again, it is crucial that at least one appears, so that we do not make a false conclusion.

All these considerations give rise to the following definition.

Definition 38. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Rein1} for S on $\mathcal{A}$ if and only if

- $S \cap X \subseteq \mathscr{A}' \subseteq X$
- $\mathscr{R}' = \varnothing$
- If $(\mathscr{A} \setminus S) \cap X \neq \varnothing$, then $\exists a \in (\mathscr{A} \setminus S) \cap X$ with $a \in \mathscr{A}'$

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$ by $Expl_{Rein1}(S)$.

Example. Consider the Argumentation Frameworks of Figures 2.1 and 2.3, and any set of arguments in those frameworks. The explanation on why this set contains all the unattacked arguments would be the empty graph for either of these frameworks. Indeed, notice that neither of them contains any unattacked argument.

Once an answer to Q^{Ext}_{Rein1} is provided, the conformity check simply consists in verifying whether or not every argument displayed belongs to S.

Notation. In the following, considering a set of arguments S and an explanation for *Rein1* for S $Expl_{Rein1}(S)$, we denote by C_{Reins1} the condition "all the arguments of $Expl_{Rein1}(S)$ belong to S".

Explanation for *Rein2*

In the case of *Rein2*, the idea is to have a subgraph in which we make sure that, if an argument, for which the questioned set attacks all the attackers, does not belong to the set, then such an argument appears in the subgraph. Indeed, the presence of such an argument is enough to conclude that there exists an argument that is acceptable with respect to the set and does not belong to it. To get these arguments, we may consider those for which the set attacks at least one attacker, in other terms, the arguments that the set defends. These are the arguments reachable in two steps of the successor function starting from the set. We should then also consider all the attackers of these arguments.

Note. In this situation, we may focus on the arguments of the questioned set, the arguments that it defends, and the attackers of those arguments. Additionally, we are only interested in attacks from the set to the attackers (to know whether or not the defended arguments are acceptable) and from the attackers to the defended arguments (to show that they are indeed attackers).

Note. We can make a similar observation as in the previous cases, concerning the attacks from the set to the attackers of the arguments it defends. For each attacker, even if only one is enough, there may very well be several of them. Just as in the previous cases, we may in this situation choose how many we keep between one and all of them, all these choices resulting in valid explanations. Yet again, it is crucial that at least one appears, so that we do not make a false conclusion.

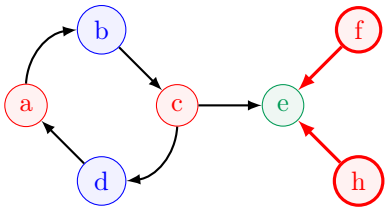

Figure 3.20: Possible explanation on why $\{b, d\}$ contains all the arguments for which it attacks all the attackers in Figure 2.2

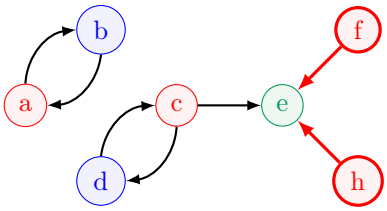

Figure 3.21: Other possible explanation on why $\{b, d\}$ contains all the arguments for which it attacks all the attackers in Figure 2.2.

All these considerations give rise to the following definition.

Definition 39. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Rein2} for S on $\mathcal{A}$ if and only if

- $A' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$
- $X \subseteq \mathscr{R}' \subseteq X \cup Y$
- For every $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a, b) \in \mathscr{R}'$ with $a \in S$

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$ by $Expl_{Rein2}(S)$.

Example. Figures 3.20 and 3.21 depict two answers to Q^{Ext}_{Rein2} for $\{b, d\}$ on the Argumentation Framework of Figure 2.2.

Once an answer to Q^{Ext}_{Rein2} is provided, the conformity check consists in verifying whether or not every argument defended by S that is not in S has an attacker that is not attacked by S. Since, by definition, in an answer to Q^{Ext}_{Rein2} these attackers can only be attacked by elements of S, this reduces to verifying whether or not every argument defended by S that is not in S has an attacker that is not attacked *at all*. In other terms, it consists in verifying whether or not every argument defended by S that is not in S has an attacker that is a source vertex.

Notation. In the following, considering a set of arguments S and an explanation for *Rein2* for S $Expl_{Rein2}(S)$, we denote by C_{Reins2} the condition "all the arguments that S defends but are not in S are attacked by a source vertex in $Expl_{Rein2}(S)$".

Explanation for Completeness

Recall that the complete semantics is made of the Coherence, Defence and Reinstatement principles, the last one being divided into the *Rein1* and *Rein2* sub-principles. As such, we can define an *answer to* Q^{Ext}_{Co} for a set S as being a set containing $Expl_{Coh}(S)$, $Expl_{Def}(S)$, $Expl_{Rein1}(S)$ and $Expl_{Rein2}(S)$.

Definition 40. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq \mathscr{A}$. An *answer to* Q^{Ext}_{Co} for S on $\mathcal{A}$ is a set $\{Expl_{Coh}(S), Expl_{Def}(S), Expl_{Rein1}(S), Expl_{Rein2}(S)\}$.

Example. Figures 3.22, 3.23, 3.17 and 3.19 constitute an answer to Q^{Ext}_{Co} on the Argumentation Framework of Figure 2.2, for $\{b\}$. In particular, Figures 3.22 and 3.19 show that $\{b\}$ respects the principles of Coherence and Defence respectively, but Figures 3.17 and 3.23 show that $\{b\}$ does not respect the *Rein1* and *Rein2* principles respectively.

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Co} for S on $\mathcal{A}$ by $Expl_{Co}(S)$.

Figure 3.22: Explanation on why $\{b\}$ is coherent in Figure 2.2

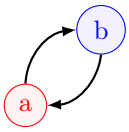

Figure 3.23: Explanation on why $\{b\}$ only contains arguments that are acceptable with respect to $\{b\}$ in Figure 2.2.

3.4.5 Explanation for Complement Attack

Recall that a set of arguments respects the Complement Attack principle if and only if all the arguments that do not belong to the set are attacked by an argument of the set. Consequently, if we are to show why a set of arguments attacks its complement, we must show a part of the graph highlighting the attacks from this set to its complement. We begin by illustrating on some examples the intuition, and then formally define these explanations.

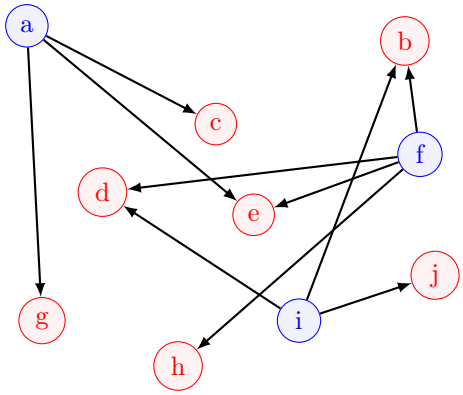

Figure 3.24: Explanation on why $\{a, f, i\}$ attacks its complement in Figure 2.3. All arguments that are not a, f or i are attacked by either a, f or i.

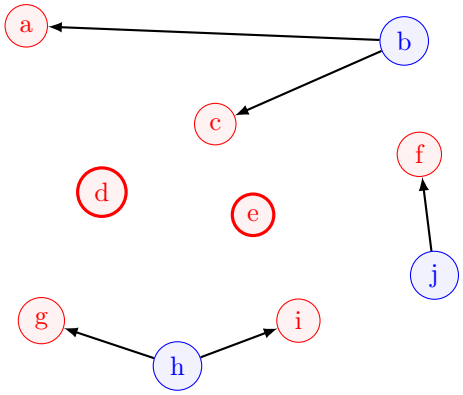

Figure 3.25: Explanation on why $\{b, h, j\}$ does not attack its complement in Figure 2.3. d and e are not attacked by b, h or j.

Example. Consider the Argumentation Framework of Figure 2.3 and the questions `"Why does {a,f,i} respect the Complement Attack principle?"` and `"Why does {b,h,j} not respect the Complement Attack principle?"`. Figures 3.24 and 3.25 show the answers for the first and second questions respectively.

The idea here is to have a subgraph in which we make sure that, for every argument that is not in the considered set, if there is at least one arc from an argument of the set to that outsider, then such an arc appears in the subgraph. Indeed, the presence of such an arc is enough to conclude that the outsider we consider is being dealt with. As a consequence, it is needed to do so for every argument that is not in the set. If no such arc exists, none is displayed and we can conclude that some outsider is being left aside, and so that the set does not respect the Complement Attack principle.

Note. In this situation, it seems best to focus both on arguments of the questioned set and the arguments that do not belong to it, that is to say, *all* the arguments. However, we are only interested in the arcs that go from the arguments of the set to those that are not in it, and so forgo all the others.

Note. Similarly to the previous cases, we can observe that, for each argument that is not in the set, even though only one arc from an argument of the set is enough, there may be several of them (for instance, on Figure 3.24, both a and f attack e). In such a case, we may again choose how many arcs we keep, between one and all of them, all these choices resulting in valid explanations. Once again, it is crucial that at least one appears, so that we do not make a false conclusion.

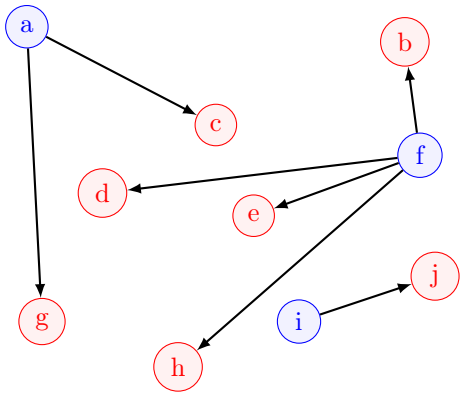

Figure 3.26: Other possible explanation on why $\{a, f, i\}$ attacks its complement in Figure 2.3.

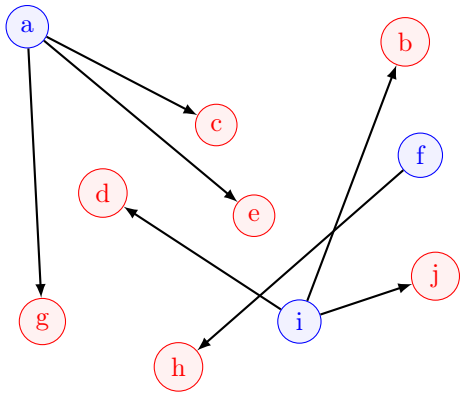

Figure 3.27: Other possible explanation on why $\{a, f, i\}$ attacks its complement in Figure 2.3.

All these considerations give rise to the following definition.

Definition 41. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{CA} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = \mathscr{A}$
- $\mathscr{R}' \subseteq X$
- $\forall b \in \mathscr{A} \setminus S$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a, b) \in \mathscr{R}'$ with $a \in S$

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$ by $Expl_{CA}(S)$.

Example. We have already seen that Figure 3.24 shows an answer to Q^{Ext}_{CA} for $\{a, f, i\}$ on the Argumentation Framework of Figure 2.3. Figures 3.26 and 3.27 depict other possible answers for the same question.

Once an answer to Q^{Ext}_{CA} is provided, the conformity check consists in verifying whether or not every argument that is not in S is attacked by S. Since, by definition, in an answer to Q^{Ext}_{CA}, arguments not in S can only be attacked by elements of S, this reduces to verifying whether or not every argument not in S is attacked *at all*. In other terms, it consists in verifying whether or not every argument not in S is not a source vertex. In addition, since the attacks from S to arguments not in S are the only attacks in the graph, it consists in fact in verifying whether or not every argument not in S is not an isolated vertex.[4]

Notation. In the following, considering a set of arguments S and an explanation for Complement Attack for S $Expl_{CA}(S)$, we denote by C_{CA} the condition "there exists no isolated vertex among the arguments that do not belong to S in $Expl_{CA}(S)$".

Explanation for Stability

Recall that the stable semantics is made of the Coherence and the Complement Attack principles. As such, we can define an *answer to* Q^{Ext}_{Sta} for a set S as being a set containing $Expl_{Coh}(S)$ and $Expl_{CA}(S)$.

Definition 42. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, and $S \subseteq \mathscr{A}$. An *answer to* Q^{Ext}_{Sta} for S on $\mathcal{A}$ is a set $\{Expl_{Coh}(S), Expl_{CA}(S)\}$.

Example. Figures 3.8 and 3.24 constitute an answer to Q^{Ext}_{CA} on the Argumentation Framework of Figure 2.3, for $\{a, f, i\}$.

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments, we will denote an answer to Q^{Ext}_{Sta} for S on $\mathcal{A}$ by $Expl_{Sta}(S)$.

[4]Note that the presence of an isolated vertex is indeed a structural property.

3.4.6 Results on Explanations for Semantics Extensions

In the previous section, we formally defined explanations for some of the different principles that constitute Abstract Argumentation semantics. We've illustrated them using examples and informally gave directions as to how to use them to explain. In particular, we have given what we called *conformity checks* that rely on *structural properties* of the explanations, so that these properties may be *seen* on the explanation graphs. As they were given in an informal way, nothing guarantees yet that these conformity checks are always valid, no matter the explanation we might consider in the explanation class of each principle. In this section, we therefore provide formal results that demonstrate the validity of each conformity check for its associated class of explanations.

Conformity checks and visual behavior

We begin with the Coherence principle. Recall that the conformity check consists in checking whether an arc exists in the explanation, the idea being that if one does, then we may conclude that the questioned set does not respect the principle. The following theorem indicates that this conformity check is correct for every explanation of the class corresponding to the Coherence principle.

Theorem 1. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Coh}(S)$ be an explanation for Coherence for S on $\mathcal{A}$. S is conflict-free if and only if $Expl_{Coh}(S)$ satisfies C_{Coh}.*

This theorem indeed states that, in order to know whether a set S of arguments is conflict-free or not (semantic property), one might just compute an explanation for Coherence $Expl_{Coh}(S)$ and see whether arcs are present or not (structural property). Interestingly, since this is an equivalence result, if we know beforehand that S is conflict-free, we can predict the appearance of $Expl_{Coh}(S)$ (i.e. being devoid of arcs).

In the case of the Defence principle, recall that the conformity check consists in verifying whether or not there exists a source vertex among the attackers of the questioned set. The following theorem indicates that this conformity check is correct for every explanation of the class corresponding to the Defence principle, provided that the questioned set is conflict-free.

Theorem 2. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ be a conflict-free set of arguments and $Expl_{Def}(S)$ be an explanation for Defence for S on $\mathcal{A}$. $S \subseteq F_{\mathcal{A}}(S)$ if and only if $Expl_{Def}(S)$ satisfies C_{Def}.*

This theorem states that, in order to know whether a conflict-free set S of arguments contains only acceptable arguments, one might just compute an explanation for Defence $Expl_{Def}(S)$ and see whether there exists a source vertex among the attackers of S or not. As we have an equivalence result, this also means that we can predict what the explanation will look like if computed on a set that we know to be admissible.

Note. The condition on the set being conflict-free to effectively use the explanation might be surprising. Recall however that an explanation for Defence is never used without an explanation for Coherence. Thus, one might want to look first at an explanation for Coherence to verify whether the questioned set is conflict-free, and if it is the case, then look at an explanation for Defence.

In addition to Theorem 2, we have an additional result concerning the visual behavior of explanations for Defence. Recall that the idea is consider a set, its attackers, and the attacks that from one to the other and vice versa. Hence, we can feel that there is an inherent separation between two groups of arguments in the explanation. The following proposition formalizes this intuition.

Proposition 4. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$ If S is conflict-free, then $Expl_{Def}(S)$ is a bipartite graph with part S.*[5][6]

As such, if an explanation for Defence is computed on a set that is conflict-free, it is possible to display it so that its vertices are separated into two groups, with the arcs always going from one group to the other. Alternatively, if one computes an explanation for Defence for a set and if it does not result in a bipartite graph with the set as one of its parts, we may conclude that the set is not conflict-free.

[5] The other part is then obviously $\mathscr{R}^{-1}(S)$

[6] S and $\mathscr{R}^{-1}(S)$ might not be the *only* possible partition of the set of arguments, but in this case it is *always* one.

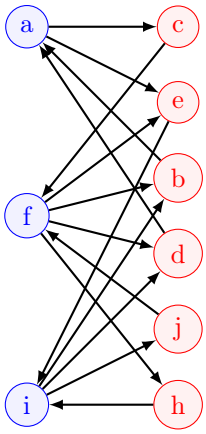

Figure 3.28: The graph of Figure 3.12 rearranged to highlight its bipartite nature

Example. Consider the explanation for Defence for $\{a, f, i\}$ on the Argumentation Framework of Figure 2.3 depicted on Figure 3.12. According to Table 2.3, $\{a, f, i\}$ is conflict-free. Thus the graph of Figure 3.12 is a bipartite graph with part $\{a, f, i\}$. Figure 3.28 shows a different display of Figure 3.12 to highlight its bipartite nature.

Regarding the Reinstatement principle, recall that we have divided it into two sub-principles: *Rein1* and *Rein2*, with their corresponding explanations. For *Rein1*, the idea is to verify that every argument displayed belongs to the questioned set. For *Rein2*, the idea is that the arguments that are defended by S but not in S must have an attacker that is a source vertex. The following theorem formalizes the correctness of both these conformity checks for the Reinstatement principle.

Theorem 3. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an explanation for Rein2 for S on $\mathcal{A}$. If $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C_{Reins2}, then $F_{\mathcal{A}}(S) \subseteq S$.*

This theorem states that by computing $Expl_{Rein1}(S)$ and verifying that all its vertices are in S, and by computing $Expl_{Rein2}(S)$ and verifying that the arguments that S defends but are not in S are all attacked by a source vertex, one verifies that S contains all the arguments that are acceptable with respect to it.

Notice that, on the contrary of previous results, this is not an equivalence result. We also have a result that reverses the implication, but it does not rely on the same conditions.

Theorem 4. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an explanation for Rein2 for S on $\mathcal{A}$. If $F_{\mathcal{A}}(S) \subseteq S$, then $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C'_{Reins2}, with C'_{Reins2} being the condition "all the arguments that S defends but are not in S are attacked by a source vertex or an argument of $\mathscr{R}^{+2}(S)$ in $Expl_{Rein2}(S)$".*

This theorem states that if we compute $Expl_{Rein1}(S)$ on a set S of arguments which we know contains all the arguments that are acceptable with respect to it, we know that all the arguments in $Expl_{Rein1}(S)$ will be contained in S. Likewise, if we compute $Expl_{Rein2}(S)$ on a similar set, we know that all the arguments that S defends but which are not in S will be attacked by a source vertex or a an argument that S defends.

From theorems 3 and 4 follows the next corollary, which shows an equivalence result for the Reinstatement principle:

Corollary 2. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ be a set of arguments such that $\mathscr{R}^{+2}(S)$ is conflict-free, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an explanation for Rein2 for S on $\mathcal{A}$. $F_{\mathcal{A}}(S) \subseteq S$ if and only if $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C_{Reins2}.*

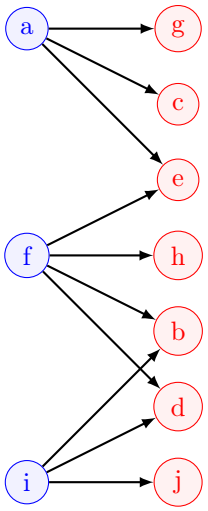

Figure 3.29: The graph of Figure 3.24 rearranged to highlight its bipartite nature

Finally, in the case of the Complement Attack principle, recall that the conformity check consists in verifying whether there is an isolated vertex among the arguments that do not belong to the questioned set. The following theorem indicates that this conformity check is correct for every explanation of the class corresponding to the Complement Attack principle.

Theorem 5. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{CA}(S)$ be an explanation for Complement Attack for S on $\mathcal{A}$. $A \setminus S \subseteq \mathscr{R}^{+1}(S)$ if and only if $Expl_{CA}(S)$ satisfies C_{CA}.*

This theorem states that, in order to know if a set S of arguments attacks its complement or not, one might just compute $Expl_{CA}(S)$ and look at the arguments not in S. If one of them is isolated, S does not attack its complement, otherwise it does. Again, since this is an equivalence result, if we compute $Expl_{CA}(S)$ on a set that we know to attack its complement, we can predict how $Expl_{CA}(S)$ will look like.

As for the Defence principle, we have an additional result regarding the visual behavior of explanations for Complement Attack. Indeed, recall that the idea is to separate the arguments between those that belong to the questioned set and those that do not, while only considering the attacks that go from the set to its complement. Again, we can sense that a clear separation between two groups of arguments can be made. The following proposition formalizes this intuition.

Proposition 5. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$. $Expl_{CA}(S)$ is a bipartite graph with part S[7][8] and every argument of S is a source vertex in $Expl_{CA}(S)$.*[9]

Thus, it is possible to display an explanation for the Complement Attack principle so that its vertices are separated in two groups, with the arcs always going from one group to the other.

Remark. Notice that Proposition 5, on the contrary of Proposition 4, does not require a condition on the set of arguments. Moreover, it is more precise in that we have an additional result on the status of the arguments in the explanation, which is not the case for Proposition 4.

Example. Consider the explanation for Complement Attack for $\{a, f, i\}$ on the Argumentation Framework of Figure 2.3 depicted on Figure 3.24. The graph of Figure 3.24 is a bipartite graph with part $\{a, f, i\}$. Figure 3.29 shows a different display of Figure 3.24 to highlight its bipartite nature.

[7] The other part is then obviously $\mathscr{A} \setminus S$.
[8] S and $\mathscr{A} \setminus S$ might not be the *only* possible partition of the set of arguments, but in this case it is *always* one.
[9] By Definition 27, this means in particular that every argument of $\mathscr{A} \setminus S$ is a sink vertex.

Properties on the classes of explanations

Now that we have formalized the correctness of the conformity checks, linking structural (and so, visual) properties to semantic ones, as well as giving additional results regarding the visual behavior of some explanations, we turn to the study of other types of properties. These additional properties help to identify specific explanations inside a given class, some of them that could be deemed undesirable, and help us understand how these classes of explanations are organised. Let us first define the properties that we investigate.

Minimality, Maximality A minimal (respectively maximal) explanation is an explanation which contains the least (respectively all the) possible amount of information. In a sense, a minimal explanation only provides what is required to explain whereas a maximal explanation in fact provides everything that might be relevant to explain, even if it might be redundant.

Definition 43. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. $Expl_\pi(S)$ is a *minimal* (respectively *maximal*) *answer to* $XVer_\pi$ if and only if there is no other $Expl_\pi(S)'$ such that $Expl_\pi(S)'$ is a strict subgraph (respectively supergraph) of $Expl_\pi(S)$.

Emptyness The notion of an empty explanation is one that should be avoided when providing explanations, in the sense that it somewhat represents the incapacity of the system to answer the question that has been asked.

Definition 44. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. $Expl_\pi(S)$ is an *empty answer to* $XVer_\pi$ if and only if $Expl_\pi(S) = (\varnothing, \varnothing)$.

Uniqueness We consider an explanation to be unique when there is only one of its kind. Although we defined classes of explanations that can represent all the different points of view that could emerge as to how to answer a question, in some situations, there can only be one way to answer that question.

Definition 45. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. $Expl_\pi(S)$ is a *unique answer to* $XVer_\pi$ if and only if there is no $Expl_\pi(S)'$ such that $Expl_\pi(S)' \neq Expl_\pi(S)$.

The first results concern empty explanations. The following theorem shows that although empty explanations can occur, they only do so in very specific situations.

Theorem 6. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$. $(\varnothing, \varnothing)$ is an answer to*

1. Q_π^{Ext} *for* S *on* $\mathcal{A}$ *with* $\pi \in \{Coh, Def, Rein2\}$ *if and only if* $S = \varnothing$.
2. Q_{Rein1}^{Ext} *for* S *on* $\mathcal{A}$ *if and only if* $\{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\} = \varnothing$.
3. Q_{CA}^{Ext} *for* S *on* $\mathcal{A}$ *if and only if* $\mathcal{A} = (\varnothing, \varnothing)$.

As such, the empty explanation is, and can only be, a valid explanation when the question Q_π^{Ext} is asked on an empty set for the principles *Coh*, *Def* and *Rein2*, and when it is asked on an empty Argumentation Framework for the principle *CA*. In the case of the principle *Rein1*, this occurs when the set of unattacked arguments in the initial Argumentation Framework is empty, which definitely less specific than for the other principles, and a lot more likely to happen in our opinion. In practice, we consider that it means that when an explanation is required for the complete semantics on an Argumentation Framework without unattacked arguments, we may skip the explanation for *Rein1* and only display the three other explanations (recall Definition 40).

Not only is the empty explanation a valid explanation in specific cases, it is itself a very specific one, as shows the following result.

Theorem 7. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. If $(\varnothing, \varnothing)$ is an answer to Q_π^{Ext} for S on $\mathcal{A}$, then it is unique.*

This theorem states that the empty explanation is in fact so specific that, when it occurs, it is the only possible explanation.

We now turn to maximal explanations. As it turns out, for every Abstract Argumentation principle that we consider, there is only one maximal explanation. This is the object of the next theorem.

Theorem 8. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. If $Expl_\pi(S)$ is a maximal explanation for π, then it is the unique maximal explanation for π.*

Since there exists only one maximal explanation for every Abstract Argumentation principle that we consider, we introduce a dedicated notation for referring to them.

Notation. In the following, considering an arbitrary Argumentation Framework $\mathcal{A}$ and given a set S of arguments and an Abstract Argumentation principle π, we will denote a maximal explanation for π for S on $\mathcal{A}$ by $MaxExpl_\pi(S)$.

We might then consider minimal explanations and wonder if a similar result exists for them as well. In general, there can be several minimal explanations for every principle.

Example. Figures 3.10 and 3.11 depict different minimal explanations for Coherence for $\{b, d, i\}$ on the Argumentation Framework of Figure 2.3.

So we do not have a uniqueness result on minimal explanations. Nevertheless, we can obtain different interesting results for them. In particular, the following lemmas give us size boundaries for minimal explanations of each explanation class.

Lemma 1. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Coh. $Expl_{Coh}(S)$ is a minimal explanation for Coh if and only if $|\mathscr{R}'| \leq 1$.*

Lemma 2. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Def. $Expl_{Def}(S)$ is a minimal explanation for Def if and only if for all $x \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Lemma 3. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Rein1. $Expl_{Rein1}(S)$ is a minimal explanation for Rein1 if and only if $|\mathscr{A}' \setminus S| \leq 1$.*

Lemma 4. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Rein2}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Rein2. $Expl_{Rein2}(S)$ is a minimal explanation for Rein2 if and only if for all $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Lemma 5. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for CA. $Expl_{CA}(S)$ is a minimal explanation for CA if and only if for all $x \notin S$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Finally, minimal and maximal explanations have a particular relation, which is the object of the next theorem.

Theorem 9. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle. Consider a maximal explanation for π $Expl_\pi(S)$, and let M be the set of all minimal explanations for π. Then $Expl_\pi(S) = \bigcup_{G \in M} G$.*

This last theorem states that, for each Abstract Argumentation principle that we consider in the present work, the maximal explanation of the class related to this principle is exactly the union (in the sense of the graph union operator defined in Definition 26) of all the minimal explanations. This gives us some insight as to how the class of explanations is organized regarding the inclusion (i.e. subgraph) relation.

3.4.7 Computing Explanations for Semantics Extensions

We now turn to how to obtain an answer to a question Q_π^{Ext} for some $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$. To do so, first recall the context in which we place ourselves (Hypothesis (H1)). Taking this into consideration, here are the elements that we consider reasonable to use to compute our explanations:

1. The initial Argumentation Framework
2. The set of arguments for which to compute the explanation
3. The Abstract Argumentation principle for which to compute the explanation

Now, using these elements, to compute explanations, we take advantage of the inner organization of classes of explanation regarding the inclusion relation (cf. Theorem 9). The idea is to have a way to obtain maximal explanations (which are unique in virtue of Theorem 8), and a way to obtain the other explanations from the maximal ones.

To get maximal explanations from the elements we listed above, we give them a characterization in terms of either their arguments, or their attack relation.

Lemma 6. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(a,b) \in \mathscr{R} \mid a, b \in S\}$. If $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Coh, then $X \subseteq \mathscr{R}'$.*

Lemma 7. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. If $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Def, then $Y \subseteq \mathscr{R}'$.*

Lemma 8. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. If $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Rein1, then $X \subseteq \mathscr{A}'$.*

Lemma 9. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. If $Expl_{Rein2}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Rein2, then $Y \subseteq \mathscr{R}'$.*

Lemma 10. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(a,b) \in \mathscr{R} \mid a \in S, b \notin S\}$. If $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for CA, then $X \subseteq \mathscr{R}'$.*

Using these Lemmas in combination with Definitions 34, 36, 38, 39 and 41, we see that we can easily and efficiently obtain maximal explanations only using the induced subgraph and spanning subgraph operators (Definitions 24 and 25) on the initial Argumentation Framework. Some maximal explanations require both operators (using the induced subgraph first, then the spanning subgraph), while others require only the use of one of them. In particular, maximal explanations for Coherence and *Rein1* are only induced subgraphs, while the maximal explanation for Complement Attack is only a spanning subgraph.

Next, from the maximal explanations, we give algorithms to compute minimal explanations. These algorithms all follow the same schema. The idea is to start from a maximal explanation and to gradually remove elements until a minimal explanation is obtained. We give one algorithm for each Abstract Argumentation principle that we consider, and name them Alg_π for $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$.

Alg_{Coh} Computation of a minimal answer to Q_{Coh}^{Ext}

Require: $\mathcal{A} = (\mathscr{A}, \mathscr{R}), S \subseteq \mathscr{A}$
1: $(\mathscr{A}', \mathscr{R}') \leftarrow MaxExpl_{Coh}(S)$
2: **while** $|\mathscr{R}'| > 1$ **do**
3: $\quad (x, y) \leftarrow choose(\mathscr{R}')$
4: $\quad \mathscr{R}' \leftarrow \mathscr{R}' \setminus \{(x, y)\}$
5: **end while**
6: **return** $(\mathscr{A}', \mathscr{R}')$

The following theorem states that our algorithms are sound and complete for the computation of minimal explanations.

Alg_{Def} Computation of a minimal answer to Q^{Ext}_{Def}

Require: $\mathcal{A} = (\mathscr{A}, \mathscr{R}), S \subseteq \mathscr{A}$
1: $(\mathscr{A}', \mathscr{R}') \leftarrow MaxExpl_{Def}(S)$
2: **for** $y \in \mathscr{R}^{-1}(S) \setminus S$ **do**
3: **while** $|\mathscr{R}'^{-1}(y)| > 1$ **do**
4: $x \leftarrow choose(\mathscr{R}'^{-1}(y))$
5: $\mathscr{R}' \leftarrow \mathscr{R}' \setminus \{(x, y)\}$
6: **end while**
7: **end for**
8: **return** $(\mathscr{A}', \mathscr{R}')$

Alg_{Rein1} Computation of a minimal answer to Q^{Ext}_{Rein1}

Require: $\mathcal{A} = (\mathscr{A}, \mathscr{R}), S \subseteq \mathscr{A}$
1: $(\mathscr{A}', \mathscr{R}') \leftarrow MaxExpl_{Rein1}(S)$
2: **while** $|\mathscr{A}' \setminus S| > 1$ **do**
3: $x \leftarrow choose(\mathscr{A}' \setminus S)$
4: $\mathscr{A}' \leftarrow \mathscr{A}' \setminus \{x\}$
5: **end while**
6: **return** $(\mathscr{A}', \mathscr{R}')$

Theorem 10. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$. Algorithm Alg_π using $\mathcal{A}$ and S as inputs is sound and complete for the computation of a minimal explanation for π.*

To finish with, we would like to point out that Algorithms Alg_π can easily be adapted to compute not a minimal explanation, but an *intermediate* one, which is neither minimal, nor maximal. To do so, it suffices to stop the removal process before the condition of the associated **While** instruction is reached. By arbitrarily doing so, we can obtain any intermediate explanation. As such, we have a process to obtain *any* explanation, given an initial Argumentation Framework, a set of arguments and an Abstract Argumentation principle among $\{Coh, Def, Rein1, Rein2, CA\}$.

3.5 Visual Explanations for Extension Membership

In the previous section (3.4), we provided explanations for some Abstract Argumentation semantics. Recall that, relying on Hypothesis (H2), we did so by providing answers to questions asked by some user, these questions being, for some set of arguments S and some semantics σ: `"Why is` S `[not] a` σ `extension?"`.

Alg_{Rein2} Computation of a minimal answer to Q^{Ext}_{Rein2}

Require: $\mathcal{A} = (\mathscr{A}, \mathscr{R}), S \subseteq \mathscr{A}$
1: $(\mathscr{A}', \mathscr{R}') \leftarrow MaxExpl_{Rein2}(S)$
2: **for** $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$ **do**
3: **while** $|\mathscr{R}'^{-1}(y)| > 1$ **do**
4: $x \leftarrow choose(\mathscr{R}'^{-1}(y))$
5: $\mathscr{R}' \leftarrow \mathscr{R}' \setminus \{(x, y)\}$
6: **end while**
7: **end for**
8: **return** $(\mathscr{A}', \mathscr{R}')$

Alg_{CA} Computation of a minimal answer to Q_{CA}^{Ext}

Require: $\mathcal{A} = (\mathscr{A}, \mathscr{R}), S \subseteq \mathscr{A}$
1: $(\mathscr{A}', \mathscr{R}') \leftarrow MaxExpl_{CA}(S)$
2: **for** $y \in \mathscr{A} \setminus \S$ **do**
3: **while** $|\mathscr{R}'^{-1}(y)| > 1$ **do**
4: $x \leftarrow choose(\mathscr{R}'^{-1}(y))$
5: $\mathscr{R}' \leftarrow \mathscr{R}' \setminus \{(x, y)\}$
6: **end while**
7: **end for**
8: **return** $(\mathscr{A}', \mathscr{R}')$

Recall as well the context, given by Hypothesis (H1), in which these questions are asked and answered:

- A contextual Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$
- A contextual semantics σ used to compute some result
- A contextual result $Res \subseteq \mathscr{A}$

Now, consider this observation: *in this particular context*, question Q_{σ}^{Ext} (in the case where $S = Res$) more or less comes down to challenging the entire result. In other terms, Q_{σ}^{Ext} requires reasons that makes S *as a whole* valid regarding semantics σ. In our context, it seems reasonable to consider that, in some situations, the user would wish to challenge, instead of the entire result, *a part of the result only*. To do so, the user would ask a question, different from Q_{σ}^{Ext}, requiring why some part of the result is in it. We call them *questions on extension membership*. Still relying on Hypothesis (H2), their answers would then be *explanations for extension membership*.

What could these questions be? We take them to be the questions: "Why is S accepted?". Note that, just like Q_{σ}^{Ext}, we can also consider the negative form of this question, that is: "Why is S not accepted?". Now, we could be tempted to interpret the term "accepted" as referring to the notion of credulous / skeptical acceptance. In our opinion, this is not the case here. Recall that, *in our context*, the user asks their question *after being presented some result*, so, most likely, *in reaction to it*. Considering this, we find more natural to interpret "Why is S accepted?" as "Why is S a part of the result?". Similarly, "Why is S not accepted?" would then be "Why is S not a part of the result?".

Remains the problem of answering these questions. What elements to give as an answer? To decide this, we must rely on what they *mean*. As we already discussed, these questions require elements that show why a set of arguments S is either part of the result or not. There could be several ways to do so. Consider the question "Why is S accepted?". To answer it, one could think of showing that S does not come at the expense of the principles that constitute the semantics σ used to compute the result *Res*. So, for instance, in the case of σ being admissibility, one could show that S is not in conflict with itself and the rest of *Res*, and that it is defended, again either by itself or the rest of *Res*. Please note that this is not exactly like showing that *Res* as a whole is admissible. It is more like doing a focus on S, keeping in mind the underlying principles of σ, here taken to be admissibility. In a sense, that would be showing that S "has the right to be in *Res*".

Following on this vision, our take is instead to take the opposing view and show that S "does not have the right not to be here". Again, please note that this is not exactly the same as what was discussed just before. Indeed, in the approach we choose, the idea is to focus on S by showing that *Res* cannot hold as a whole without it. In other terms, answering the question "Why is S accepted?" is showing that S is a *necessary part of Res*. With a similar reasoning, answering "Why is S not accepted?" is then showing that S is *incompatible with Res*. This is thus what we will aim at showing when answering these questions in the following.

We can go even further regarding questions on extension memberships. From our previous discussion, we interpret the question "Why is S accepted?" as requiring the elements that show that S is necessary in *Res*. Recall that this question is, in our context, asked in reaction to *Res* being presented to the user. From this, we would assume that *the user disagrees with S being part of the result*. Similarly, from the question "Why is S not accepted?", we could assume that *the user disagrees with S not being part of the result*. But what if the user wishes to express both disagreements at the same time?

Indeed, consider the questions "Why is S accepted and not S'?" and "Why is S not accepted and not S'?". They are slightly modified versions of the questions considered before. In particular, they are contrastive. Of course, in both cases, the contrast is made on S. That is to say, S' is brought in opposition to S. It seems clear that from the question "Why is S accepted and not S'?" we could assume that *the user disagrees with S being part of the result and S' not being part of it*. Similarly, from the question "Why is S not accepted and not S'?" we could assume that *the user disagrees with S not being part of the result and S' being part of it*.

These are thus the questions to which we will aim to provide answers in this section. We now proceed to define notations for them. To do so, we first distinguish between contrastive and non-contrastive questions. Then, we take into account the presence or the absence of a negation on the property of being "accepted". We will call "positive" the questions that refer to being "accepted" and "negative" the questions that refer to being "not accepted". So, given two sets of arguments S and S', we are interested in the following questions.

$QNCP_{\sigma}^{Mem}$ (non-constrastive positive): "Why is S accepted?"

$QNCN_{\sigma}^{Mem}$ (non-constrastive negative): "Why is S not accepted?"

QCP_{σ}^{Mem} (constrastive positive): "Why is S accepted and not S'?"

QCN_{σ}^{Mem} (constrastive negative): "Why is S not accepted and not S'?"

3.5.1 Non-contrastive Questions

We begin by addressing the case of questions in which no contrast is made, i.e. questions of the form "Why is S accepted?" and "Why is S not accepted?". Recall that we consider these questions to ask for the reasons that make S a necessary part of, or not possible to include in respectively, the result *Res*. Assuming that S is a necessary part of the result, one way to show it is to show that *Res* without S is not a valid result anymore. Likewise, assuming that S cannot be included in the result, one way to show it is to show that *Res* to which we add S is not a valid result anymore.

Positive Non-contrastive Questions

Recall that, by Hypothesis (H1), *Res* was obtained by computing an extension of some semantics σ on an Argumentation Framework. Thus, showing that *Res* without S is not a valid result anymore reduces to showing that $Res \setminus S$ is not an extension of the σ semantics in the Argumentation Framework that we consider. This is precisely what the explanations we defined in Section 3.4 are for.

Definition 46. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, $S \subseteq \mathscr{A}$ and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QNCP_{\sigma}^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_{σ}^{Ext} for $Res \setminus S$ on $\mathcal{A}$.

Note. Note that, extrapolating from Definition 46, considering an Argumentation Framework $\mathcal{A}$, a result *Res* of some semantics σ, and some set of arguments S, we may consider the question "Why is S accepted?" to be in fact the question "Why is $Res \setminus S$ not a σ extension?".

Example. Consider the Argumentation Framework of Figure 2.3 and the result $Res = \{a, f, i\}$ for the admissible semantics. Suppose the user asks the question "Why is $\{a\}$ accepted?". Figures 3.30 and 3.31 show the corresponding answer: $\{a, f, i\}$ without a is not admissible. Figure 3.30 shows that, without a, the result would still respect the Coherence principle. However, Figure 3.31 shows that, without a, the result

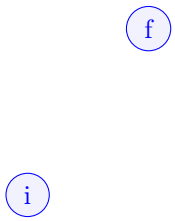

Figure 3.30: Explanation on why $\{a, f, i\}$ without a respects the Coherence principle. The conformity check is verified: there is no arc in the explanation.

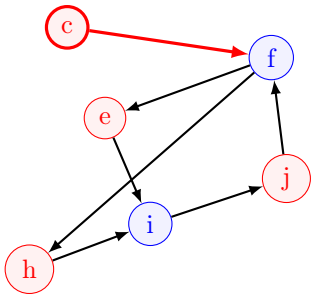

Figure 3.31: Explanation on why $\{a, f, i\}$ without a does not respect the Defence principle. The conformity check is not verified: c, a attacker of f is a source vertex in the explanation.

would not respect the Defence principle. As such, *in this particular context* (Argumentation Framework, result, semantics), a is necessary in the result presented by the system because it plays a role in how the result defends itself as a whole.

Note. Notice that, in the previous example, although we considered a question on only one argument (a), we wrote the question as if it were on the singleton set $\{a\}$. This is because questions have been introduced as being on sets in Section 3.2.1. It does not seem unreasonable however to imagine that questions may, and will, be asked on only one argument. Thus, in the following, we identify questions on a single argument x as the same question on the singleton set $\{x\}$.

Negative Non-contrastive Questions

We turn now to negative questions. The methodology is very similar as for the positive questions. Recall that we aim at showing, in this case, that S is incompatible with *Res*. This reduces to showing that $Res \cup S$ is not an extension of the σ semantics in the Argumentation Framework that we consider. Again, we can use the explanations defined in Section 3.4 to achieve this.

Definition 47. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, $S \subseteq \mathscr{A}$ and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QNCN_\sigma^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_σ^{Ext} for $Res \cup S$ on $\mathcal{A}$.

Note. Note that, extrapolating from Definition 47, considering an Argumentation Framework $\mathcal{A}$, a result *Res* of some semantics σ, and some set of arguments S, we may consider the question `"Why is` S `not accepted?"` to be in fact the question `"Why is` $Res \cup S$ `not a` σ `extension?"`.

Example. Consider the Argumentation Framework of Figure 2.3 and the result $Res = \{a, f, i\}$ for the admissible semantics. Suppose the user asks the question `"Why is` $\{d\}$ `not accepted?"`. Figures 3.32 and 3.33 show the corresponding answer: $\{a, f, i\}$ with d is not admissible. Figure 3.32 shows that, with d, the result would not respect the Coherence principle. However, Figure 3.33 shows that, with d, the result would still respect the Defence principle. As such, *in this particular context*, d cannot be included in the result presented by the system because it would add some internal conflicts.

3.5.2 Contrastive Questions

After the case of non-contrastive questions comes the case of contrastive ones. Recall that this concerns questions of the form `"Why is` S `accepted and not` S'`?"` and `"Why is` S `not accepted and not` S'`?"`. For these questions to be properly answered, we first discuss what they *mean*.

As we stated at the beginning of Section 3.5, in these questions, the contrast is of course made on the set of arguments. That is to say, in their question, the user proposes a second set (S') that is meant to be put

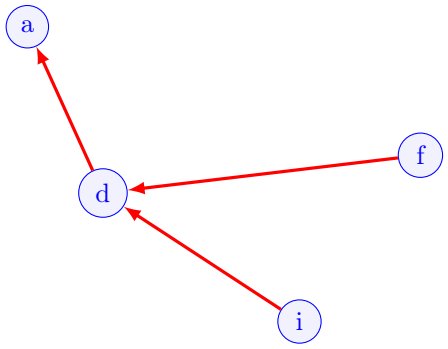

Figure 3.32: Explanation on why $\{a, f, i\}$ with d does not respect the Coherence principle. The conformity check is not verified: there are some arcs in the explanation.

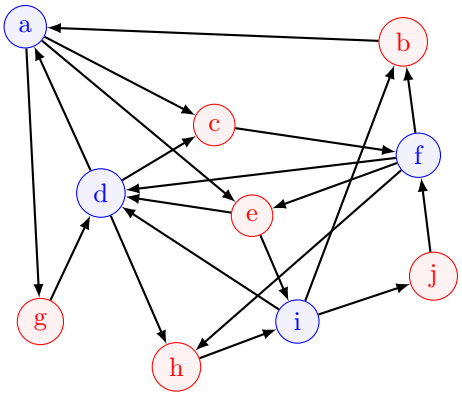

Figure 3.33: Explanation on why $\{a, f, i\}$ without a does not respect the Defence principle. The conformity check is verified: no attacker of a, d, f or i is a source vertex in the explanation.

in contrast with the first one (S). What is critically important is to understand correctly the effect of this contrast. It seems clear that, in these questions, S and S' are put in opposition. But regarding what? We argue that this is regarding their status of being "accepted" or not. Recall the contextual information that is available: an Argumentation Framework, a semantics, and an extension computed using the semantics. There is no indication in the question that S' is to be put in relation to different contextual information. Hence, we can *implicitly* infer they should be the same as for S. So, the only aspect left in order to oppose S and S' is their acceptability status, which is made implicitly. Making this opposition explicit, a question like "Why is S accepted and not S'?" is to be understood as "Why is S part of the result and S' not part of the result?" (implicitly, for the same result, of the same semantics on the same Argumentation Framework). Using our terminology, that would be the question "Why is S accepted and not S' accepted?". And the second question "Why is S not accepted and not S'?" would then become "Why is S not accepted and not S' not accepted?".

Of course these reformulations seem way less natural. We consider this to be perfectly normal since, as we said, they make explicit some information that does not need to be to understand the question. We as humans, are probably not accustomed to that way of doing. In the case of the second question, this also introduces a new negation, which then makes the whole question feels like it has too many of them. Nonetheless, we can understand that these reformulations are indeed the same question. Moreover, since they uncover some information that was previously left implicit, we would say that they give precious insight as to how these questions are built, and how they should be understood. Indeed, notice how the reformulations are in fact made of two statements, identically structured, concatenated by the connector "and not". From this observation, we deduce that the question "Why is S accepted and not S' accepted?" is in fact the concatenation of two questions, that is "Why is S accepted and why is S' not accepted?". And the question "Why is S not accepted and not S' not accepted?" would then be "Why is S not accepted and why is S' not not accepted?", or by eliminating the double negation, "Why is S not accepted and why is S' accepted?". Importantly, in both cases, the two questions that are concatenated are non-contrastive questions for extension membership, for which we already have answers available. Thus, we will use these observations to answer to the contrastive questions, following a similar line of thoughts as for how non-contrastive questions are answered.

Note. Note that, since contrastive questions are made of two statements with identical structure, and in which the set of arguments can be either "accepted" or "not accepted", from a combinatorial perspective, there are four possible contrastive questions.

For the sake of being exhaustive, we will consider all four possibilities, even though only two initially

motivated us. We will name them following the same process as at the beginning of Section 3.5. The only difference is that, since we handle two statements, we should precise for *both of them* if they are either positive or negative. Note that since the connector used to concatenate the two statements is "and not", the property of being accepted or not in the second statement should be understood as being reversed. Thus, considering two sets of arguments S and S', we obtain the following questions:

- Positive-positive contrastive question for extension membership $QCPP_{\sigma}^{Mem}$: "Why is S accepted and not S' not accepted?"
- Positive-negative contrastive question for extension membership $QCPN_{\sigma}^{Mem}$: "Why is S accepted and not S' accepted?"
- Negative-positive contrastive question for extension membership $QCPP_{\sigma}^{Mem}$: "Why is S not accepted and not S' not accepted?"
- Negative-negative contrastive question for extension membership $QCPP_{\sigma}^{Mem}$: "Why is S not accepted and not S' accepted?"

Note. Recall that $QCPN_{\sigma}^{Mem}$ corresponds in fact to our initial question QCP_{σ}^{Mem}, and that $QCNP_{\sigma}^{Mem}$ corresponds in fact to our initial question QCN_{σ}^{Mem}.

Although we consider that each of our constrastive questions has the same meaning as a concatenation of two non-constrastive questions, we think that it is not correct to answer them using the sequence of individual answers to the non-contrastive questions. Instead, we consider that concatenating the two non-contrastive questions, and not expressing them independently, is an invitation to providing *one single answer* for *both of them* simultaneously. This is thus what we shall aim for.

Positive-Positive Contrastive Questions

We begin with positive-positive contrastive questions. Along the line of thoughts of Section 3.5.1, an answer for the question "Why is S accepted and not S' not accepted?" should show simultaneously that *Res* without S is not a valid result anymore, but also that *Res* without S' is not a valid result anymore. Hence, such an answer should show that $(Res \setminus S) \setminus S'$ is not an extension of the σ semantics used to compute *Res* in the Argumentation Framework that is considered.

Definition 48. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, S, $S' \subseteq \mathscr{A}$, and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QCPP_{\sigma}^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_{σ}^{Ext} for $(Res \setminus S) \setminus S'$ on $\mathcal{A}$.

Note. Note that $(Res \setminus S) \setminus S' = Res \setminus (S \cup S')$. As such, the question "Why is S accepted and not S' not accepted?" is in fact the question "Why is $S \cup S'$ accepted?", which in turn becomes the question "Why is $Res \setminus (S \cup S')$ not a σ-extension?". This may explain why the question $QCPP_{\sigma}^{Mem}$ for S and S' on $\mathcal{A}$ and *Res* seems quite unnatural: there exists a much simpler way to ask the same question, that is question $QNCP_{\sigma}^{Mem}$ for $S \cup S'$ on $\mathcal{A}$ and *Res*.

Remark. Notice as well that, by eliminating double negations, the question "Why is S accepted and not S' not accepted?" becomes the more natural question "Why is S accepted and S' accepted?"

Example. Consider the Argumentation Framework of Figure 2.2 and the result $Res = \{a, c, f, i\}$ for the stable semantics. Suppose the user asks the question "Why is $\{f\}$ accepted and not $\{i\}$ not accepted?". Figures 3.34 and 3.35 show the corresponding answer: $\{a, c\}$ without f and without i is not stable. Figure 3.34 shows that, without f and without i, the result would still respect the Coherence principle. However, Figure 3.35 shows that, without f and without i, the result would not respect the Complement Attack principle. As such, *in this particular context*, f and i are necessary in the result presented by the system because they play a role in how the result attacks all the arguments that do not belong to it.

Figure 3.34: Explanation on why $\{a, c, f, i\}$ without f and without i respects the Coherence principle. The conformity check is verified: there is no arc in the explanation.

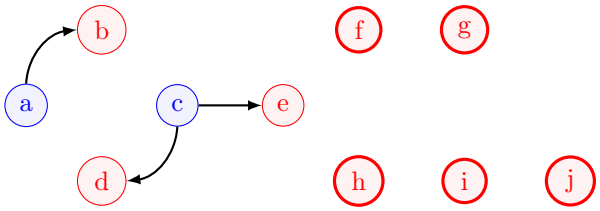

Figure 3.35: Explanation on why $\{a, c, f, i\}$ without f and without i does not respect the Complement Attack principle. The conformity check is not verified: f, g, h, i and j, which are not part of $\{a, c\}$ are isolated.

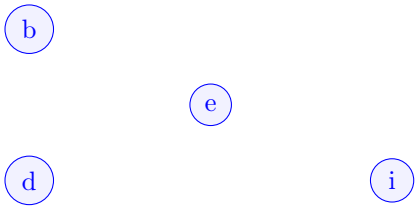

Figure 3.36: Explanation on why $\{b, d, f, i\}$ without f but adding e respects the Coherence principle. The conformity check is verified: there is no arc in the explanation.

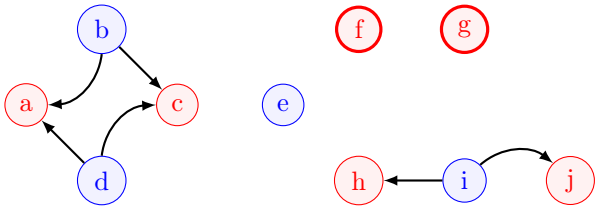

Figure 3.37: Explanation on why $\{b, d, f, i\}$ without f but adding e does not respect the Complement Attack principle. The conformity check is not verified: f and g, which are not part of $\{b, d, e, i\}$ are isolated.

Positive-Negative Contrastive Questions

Next is the case of positive-negative contrastive questions. Following the same methodology as in the previous case, an answer to "Why is S accepted and not S' accepted?" should show simultaneously that *Res* without S is not a valid result anymore, but also that *Res* to which we add S' is not a valid result anymore. Hence, such an answer should show that $(Res \setminus S) \cup S'$ is not an extension of the σ semantics used to compute *Res* in the Argumentation Framework that is considered.

Definition 49. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, S, $S' \subseteq \mathscr{A}$, and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QCPN_\sigma^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_σ^{Ext} for $(Res \setminus S) \cup S'$ on $\mathcal{A}$.

Note. Recall that the question $QCPN_\sigma^{Mem}$ "Why is S accepted and not S' accepted?" is in fact the question QCP_σ^{Mem} "Why is S accepted and not S'?" that initially motivated us.

Example. Consider the Argumentation Framework of Figure 2.2 and the result $Res = \{b, d, f, i\}$ for the stable semantics. Suppose the user asks the question "Why is $\{f\}$ accepted and not $\{e\}$?". Figures 3.36 and 3.37 show the corresponding answer: $\{b, d, f, i\}$ without f and adding e is not stable. Figure 3.36 shows that, without f and adding e, the result would still respect the Coherence principle. However, Figure 3.37 shows that, without f and adding e, the result would not respect the Complement Attack principle. As such, *in this particular context*, f is necessary in the result presented by the system because it plays a role in how the result attacks all the arguments that do not belong to it, and e cannot be used to replace f.

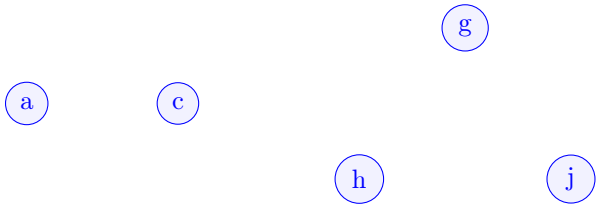

Figure 3.38: Explanation on why $\{a, c, f, h, j\}$ adding g but without f respects the Coherence principle. The conformity check is verified: there is no arc in the explanation.

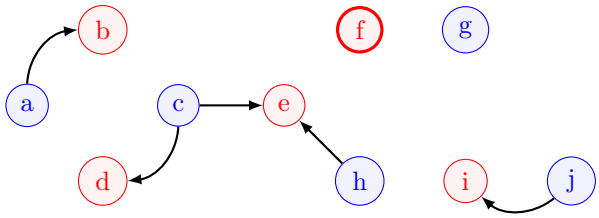

Figure 3.39: Explanation on why $\{a, c, f, h, i\}$ adding g but without f does not respect the Complement Attack principle. The conformity check is not verified: f, which is not part of $\{a, c, g, h, i\}$ is isolated.

Negative-Positive Contrastive Questions

Then, comes the case of negative-positive contrastive questions. Following the same methodology as in the previous cases, an answer to `"Why is` S `not accepted and not` S' `not accepted?"` should show simultaneously that *Res* to which we add S is not a valid result anymore, but also that *Res* without S' is not a valid result anymore. Hence, such an answer should show that $(Res \cup S) \setminus S'$ is not an extension of the σ semantics used to compute *Res* in the Argumentation Framework that is considered.

Definition 50. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, S, $S' \subseteq \mathscr{A}$, and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QCNP_\sigma^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_σ^{Ext} for $(Res \cup S) \setminus S'$ on $\mathcal{A}$.

Note. Recall that the question $QCNP_\sigma^{Mem}$ `"Why is` S `not accepted and not` S' `not accepted?"` is in fact the question QCN_σ^{Mem} `"Why is` S `not accepted and not` S'`?"` that initially motivated us.

Remark. Notice as well that, by eliminating double negations, the question `"Why is` S `not accepted and not` S' `not accepted?"` becomes the more natural question `"Why is` S `not accepted and` S' `accepted?"`

Example. Consider the Argumentation Framework of Figure 2.2 and the result $Res = \{a, c, f, h, j\}$ for the stable semantics. Suppose the user asks the question `"Why is` $\{g\}$ `not accepted and not` $\{f\}$`?"`. Figures 3.38 and 3.39 show the corresponding answer: $\{a, c, f, h, j\}$ adding g but without f is not stable. Figure 3.38 shows that, adding g but without f, the result would still respect the Coherence principle. However, Figure 3.39 shows that, adding g but without f, the result would not respect the Complement Attack principle. As such, *in this particular context*, g is not enough to replace f in its role of attacking arguments that do not belong to the result.

Negative-Negative Contrastive Questions

Finally, we treat the case of negative-negative contrastive questions. Following the same methodology as in the previous cases, an answer to `"Why is` S `not accepted and not` S' `accepted?"` should show simultaneously that *Res* to which we add S is not a valid result anymore, but also that *Res* to which we add S' is not a valid result anymore. Hence, such an answer should show that $(Res \cup S) \cup S'$ is not an extension of the σ semantics used to compute *Res* in the Argumentation Framework that is considered.

Definition 51. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, *Res*, S, $S' \subseteq \mathscr{A}$, and $\sigma \in \{CF, Adm, Co, Sta\}$ be an Abstract Argumentation semantics. An *answer to* $QCNN_\sigma^{Mem}$ for S on $\mathcal{A}$ and *Res* is an answer to Q_σ^{Ext} for $(Res \cup S) \cup S'$ on $\mathcal{A}$.

Note. Note that $(Res \cup S) \cup S' = Res \cup (S \cup S')$. As such, the question `"Why is` S `not accepted and not` S' `accepted?"` is in fact the question `"Why is` $S \cup S'$ `not accepted?"`, which in turn becomes the question

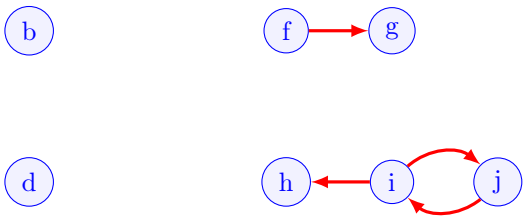

Figure 3.40: Explanation on why $\{b, d, f, h, j\}$ adding g and adding i does not respect the Coherence principle. The conformity check is not verified: some arcs are present in the explanation.

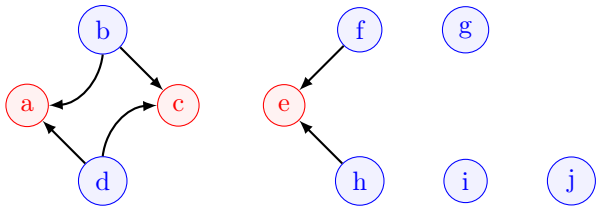

Figure 3.41: Explanation on why $\{b, d, f, h, j\}$ adding g and adding i respects the Complement Attack principle. The conformity check is verified: none of the arguments that are not part of $\{b, d, f, g, h, i, j\}$ is isolated.

`"Why is` $Res \cup (S \cup S')$ `not a` σ`-extension?"`. As for the question $QCPP_{\sigma}^{Mem}$, this may explain why the question $QCNN_{\sigma}^{Mem}$ for S and S' on $\mathcal{A}$ and Res seems quite unnatural: there exists a much simpler way to ask the same question, that is question $QNCN_{\sigma}^{Mem}$ for $S \cup S'$ on $\mathcal{A}$ and Res.

Example. Consider the Argumentation Framework of Figure 2.2 and the result $Res = \{b, d, f, h, j\}$ for the stable semantics. Suppose the user asks the question `"Why is` $\{g\}$ `not accepted and not` $\{i\}$ `accepted?"`. Figures 3.40 and 3.41 show the corresponding answer: $\{b, d, f, h, j\}$ adding g and adding i is not stable. Figure 3.40 shows that, adding g and adding i, the result would not respect the Coherence principle. However, Figure 3.41 shows that, adding g and adding i, the result would still respect the Complement Attack principle. As such, *in this particular context*, g and i cannot be added to the result, because it leads to internal conflicts.

3.6 Summary

In this section, we summarize the different elements of our approach on explanations for Abstract Argumentation. We recall the most important notions and how they are related. At the end of the section, we present a recap example of how we envision our explanations to be used.

3.6.1 Questions and Explanations

First of all, we recall our hypotheses:

(H1)**:** *A user asks for an explanation after they have been presented the result of an Abstract Argumentation selection process by some program.*

(H2)**:** *An explanation is an answer to some question.*

(H3)**:** *The user has no expert knowledge of Abstract Argumentation.*

Then, let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be the Argumentation Framework that has been used to compute some set of arguments $Res \subseteq \mathscr{A}$ using a semantics $\sigma \in \{CF, Adm, Co, Sta\}$. The questions we are interested in are:

Questions on Abstract Argumentation semantics: let S be a set of arguments.

Q_{σ}^{Ext} : `"Why is S [not] a` σ `extension?"`

To answer this question, we use some questions related to the principles behind the semantics (Coherence, Defence, Reinstatement (itself divided into two sub-principles *Rein1* and *Rein2*) and Complement Attack).

Questions on Abstract Argumentation principles: let S be a set of arguments and $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$ be an Abstract Argumentation principle.

Q_{π}^{Ext} : `"Why does S [not] respect the` π `principle?"`

The answer to Q_{σ}^{Ext} is then a set of answers to Q_{π}^{Ext}, one for each principle π that composes the semantics σ.

Questions on extension membership: let S, S' be sets of arguments.

$QNCP_{\sigma}^{Mem}$: `Why is` S `accepted?` (positive non-constrastive question)

$QNCN_{\sigma}^{Mem}$: `Why is` S `not accepted?` (negative non-constrastive question)

$QCPP_{\sigma}^{Mem}$: `Why is` S `accepted and not` S' `not accepted?` (positive-positive constrastive question)

$QCPN_{\sigma}^{Mem}$: `Why is` S `accepted and not` S' `accepted?` (positive-negative constrastive question);[10]

$QCNP_{\sigma}^{Mem}$: `Why is` S `not accepted and not` S' `not accepted?` (negative-positive constrastive question);[11]

$QCNN_{\sigma}^{Mem}$: `Why is` S `not accepted and not` S' `accepted?` (negative-negative constrastive question)

The answers to questions Q_{π}^{Ext} (instanciated for each principle π) are given in Table 3.1. The answers to questions Q_{σ}^{Ext} (instanciated for each semantics σ) are given in Table 3.2. Finally, the answers to questions $QNCP_{\sigma}^{Mem}$, $QNCN_{\sigma}^{Mem}$, $QCPP_{\sigma}^{Mem}$, $QCPN_{\sigma}^{Mem}$, $QCNP_{\sigma}^{Mem}$ and $QCNN_{\sigma}^{Mem}$ are given in Table 3.3

As noted in Section 3.5.1, some contrastive questions are equivalent to non-contrastive ones:

- $QCPP_{\sigma}^{Mem}$: `Why is` S `accepted and not` S' `not accepted?` is equivalent to $QNCP_{\sigma}^{Mem}$: `Why is` $S \cup S'$ `accepted?`.
- $QCNN_{\sigma}^{Mem}$: `Why is` S `not accepted and not` S' `accepted?` is equivalent to $QNCN_{\sigma}^{Mem}$: `Why is` $S \cup S'$ `not accepted?`.

From these equivalences, it follows that S and S' can be swapped in $QCPP_{\sigma}^{Mem}$ and $QCNN_{\sigma}^{Mem}$ without any consequence on the answer which is provided. In other terms, fact and foil are treated equivalently in these questions. This can lead to a little simplification of our approach, considering only two kinds of contrastive questions, $QCPN_{\sigma}^{Mem}$ and $QCNP_{\sigma}^{Mem}$ ($QCPP_{\sigma}^{Mem}$ and $QCNN_{\sigma}^{Mem}$ being transformed into non-contrastive questions).

Regarding these two remaining questions, a specific case can lead to another simplification: if $S \cap S' = \varnothing$, then $QCPN_{\sigma}^{Mem}$ and $QCNP_{\sigma}^{Mem}$ can lead to equivalent reformulations with an identical answer, *if the positions of S and S' are swapped between the questions.* Indeed, in this case, $(Res \setminus S) \cup S' = (Res \cup S') \setminus S$, so the answers to $QCPN_{\sigma}^{Mem}$ and $QCNP_{\sigma}^{Mem}$ are the same, and so the questions can be deemed equivalent. Thus, in this situation, considering only either $QCPN_{\sigma}^{Mem}$ or $QCNP_{\sigma}^{Mem}$ could be sufficient.

Finally, the formulation of the contrastive questions that we choose imposed "and not" as a connector to introduce the contrastive part, thus adding a new negation into the question. Questions $QCPP_{\sigma}^{Mem}$ and $QCNP_{\sigma}^{Mem}$ could then be reformulated in a more natural by eliminating double negations: $QCPP_{\sigma}^{Mem}$: `Why is` S `accepted and` S' `accepted?`, $QCNP_{\sigma}^{Mem}$: `Why is` S `not accepted and` S' `accepted?`

[10] Recall that this question corresponds to the simple contrastive positive question QCP_{σ}^{Mem}: `Why is` S `accepted and not` S'`?`

[11] Recall that this question corresponds to the simple contrastive negative question QCN_{σ}^{Mem}: `Why is` S `not accepted and not` S'`?`

Question on Coherence Q^{Ext}_{Coh}: `Why does` S `respect the Coherence principle?` Answer : Explanation $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ (see Definition 34)	
with $(\mathscr{A}', \mathscr{R}')$ a subgraph defined by:	Considering $X = \{(a,b) \in \mathscr{R} \mid a, b \in S\}$: • $\mathscr{A}' = S$ • $\mathscr{R}' \subseteq X$ • If $X \neq \varnothing$, then $\mathscr{R}' \neq \varnothing$
Conformity check:	C_{Coh}: there exists no arcs in $Expl_{Coh}(S)$ (see Theorem 1)
Question on Defence Q^{Ext}_{Def}: `Why does` S `respect the Defence principle?` Answer : Explanation $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ (see Definition 36)	
with $(\mathscr{A}', \mathscr{R}')$ a subgraph defined by:	Considering $X = \{(b,a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$: • $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$ • $X \subseteq \mathscr{R}' \subseteq X \cup Y$ • $\forall b \in \mathscr{R}^{-1}(S)$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a,b) \in \mathscr{R}'$ with $a \in S$
Conformity check:	C_{Def}: there exists no source vertex among the attackers of S in $Expl_{Coh}(S)$ (see Theorem 2)
Question on *Rein1* Q^{Ext}_{Rein1}: `Why does` S `respect the` *Rein1* `principle?` Answer : Explanation $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ (see Definition 38)	
with $(\mathscr{A}', \mathscr{R}')$ a subgraph defined by:	Considering $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$: • $S \cap X \subseteq \mathscr{A}' \subseteq X$ • $\mathscr{R}' = \varnothing$ • If $(\mathscr{A} \setminus S) \cap X \neq \varnothing$, then $\exists a \in (\mathscr{A} \setminus S) \cap X$ with $a \in \mathscr{A}'$
Conformity check:	C_{Reins1}: all the arguments of $Expl_{Rein1}(S)$ belong to S (see Theorems 3 and 4)
Question on *Rein2* Q^{Ext}_{Rein2}: `Why does` S `respect the` *Rein2* `principle?` Answer : Explanation $Expl_{Rein2}(S) = (\mathscr{A}', \mathscr{R}')$ (see Definition 39)	
with $(\mathscr{A}', \mathscr{R}')$ a subgraph defined by:	Considering $X = \{(b,c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$: • $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ • $X \subseteq \mathscr{R}' \subseteq X \cup Y$ • For every $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a,b) \in \mathscr{R}'$ with $a \in S$
Conformity check:	C_{Reins2}: all the arguments that S defends but are not in S are attacked by a source vertex in $Expl_{Rein2}(S)$ (see Theorems 3) C'_{Reins2}: all the arguments that S defends but are not in S are attacked by a source vertex or an argument of $\mathscr{R}^{+2}(S)$ in $Expl_{Rein2}(S)$ (see Theorems 4)
Question on Complement Attack Q^{Ext}_{CA}: `Why does` S `respect the Complement Attack principle?` Answer : Explanation $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ (see Definition 41)	
with $(\mathscr{A}', \mathscr{R}')$ a subgraph defined by:	Considering $X = \{(a,b) \in \mathscr{R} \mid a \in S, b \notin S\}$: • $\mathscr{A}' = \mathscr{A}$ • $\mathscr{R}' \subseteq X$ • $\forall b \in \mathscr{A} \setminus S$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a,b) \in \mathscr{R}'$ with $a \in S$
Conformity check:	C_{CA}: there exists no isolated vertex among the arguments that do not belong to S in $Expl_{CA}(S)$ (see Theorem 5)

Table 3.1: Questions on Abstract Argumentation principles and their answers, considering an Argumentation Framework $\mathcal{A}$ and a principle $\pi \in \{Coh, Def, Rein1, Rein2, CA\}$

Question on Conflict-freeness Q^{Ext}_{CF}: `Why is` S `a conflict-free extension?`	
Answer : Explanation $Expl_{CF}(S)$ = a set of subgraphs (see Definition 35)	
Subgraphs:	$\{Expl_{Coh}(S)\}$
Conformity checks:	C_{Coh} applied on $Expl_{Coh}(S)$
Question on Admissibility Q^{Ext}_{Adm}: `Why is` S `an admissible extension?`	
Answer : Explanation $Expl_{Adm}(S)$ = a set of subgraphs (see Definition 37)	
Subgraphs:	$\{Expl_{Coh}(S), Expl_{Def}(S)\}$
Conformity checks:	C_{Coh} applied on $Expl_{Coh}(S)$, C_{Def} applied on $Expl_{Def}(S)$
Question on Completeness Q^{Ext}_{Adm}: `Why is` S `a complete extension?`	
Answer : Explanation $Expl_{Co}(S)$ = a set of subgraphs (see Definition 40)	
Subgraphs:	$\{Expl_{Coh}(S), Expl_{Def}(S), Expl_{Rein1}(S), Expl_{Rein2}(S)\}$
Conformity checks:	C_{Coh} applied on $Expl_{Coh}(S)$, C_{Def} applied on $Expl_{Def}(S)$, C_{Reins1} applied on $Expl_{Rein1}(S)$, C_{Reins2} (or C'_{Reins2}) applied on $Expl_{Rein2}(S)$
Question on Stability Q^{Ext}_{Sta}: `Why is` S `a stable extension?`	
Answer : Explanation $Expl_{Sta}(S)$ = a set of subgraphs (see Definition 42)	
Subgraphs:	$\{Expl_{Coh}(S), Expl_{CA}(S)\}$
Conformity checks:	C_{Coh} applied on $Expl_{Coh}(S)$, C_{CA} applied on $Expl_{CA}(S)$

Table 3.2: Questions on Abstract Argumentation semantics and their answers, considering an Argumentation Framework $\mathcal{A}$ and a semantics $\sigma \in \{CF, Adm, Co, Sta\}$

$QNCP^{Mem}_{\sigma}$: `Why is` S `accepted?`
Answer : Explanation $Expl_{\sigma}(Res \setminus S)$ (see Definition 46)
$QNCN^{Mem}_{\sigma}$: `Why is` S `not accepted?`
Answer : Explanation $Expl_{\sigma}(Res \cup S)$ (see Definition 47)
$QCPP^{Mem}_{\sigma}$: `Why is` S `accepted and not` S' `not accepted?`
Answer : Explanation $Expl_{\sigma}((Res \setminus S) \setminus S')$ (see Definition 48)
$QCPN^{Mem}_{\sigma}$ (and QCP^{Mem}_{σ}): `Why is` S `accepted and not` S' `accepted?`
Answer : Explanation $Expl_{\sigma}((Res \setminus S) \cup S')$ (see Definition 49)
$QCNP^{Mem}_{\sigma}$ (and QCN^{Mem}_{σ}): `Why is` S `not accepted and not` S' `not accepted?`
Answer : Explanation $Expl_{\sigma}((Res \cup S) \setminus S')$ (see Definition 50)
$QCNN^{Mem}_{\sigma}$: `Why is` S `not accepted and not` S' `accepted?`
Answer : Explanation $Expl_{\sigma}((Res \cup S) \cup S')$ (see Definition 51)

Table 3.3: Questions on Abstract Argumentation principles and their answers, considering an Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, a result $Res \subseteq \mathscr{A}$ and a semantics $\sigma \in \{CF, Adm, Co, Sta\}$

3.6.2 Recap Example

In this section, we provide an example which summarizes the entire approach. Suppose that a political debate is held. Two candidates, that we will soberly name Candidate 1 and Candidate 2, make propositions on some defined arbitrary subject and confront them. The propositions of each candidate are given in Table 3.4. A computer program (the system) is used to extract arguments from their statements, and identify the conflicts between them. In more formal terms, the system creates an Argumentation Framework that supposedly represents the debate. The Argumentation Framework obtained corresponds to Figure 3.42. We will thus consider it to be our contextual Argumentation Framework which we denote $\mathcal{A} = (\mathscr{A}, \mathscr{R})$.

From this representation, the objective is then to identify some propositions that can collectively act as a viable outcome of the debate. In more formal terms, this would correspond to selecting an extension from some semantics. We will assume that the conditions under which the outcome is selected has been given beforehand. Suppose that we want some strong propositions that represent a radical point of view by defeating all the other propositions that were given. This would correspond to the dominant notion of a stable extension. Thus, we consider the stable semantics to be our contextual semantics, that we denote σ. In the Argumentation Framework of Figure 3.42, such an extension would be $\{b, c, g, i, l, p\}$.

We consider that this is indeed the outcome that is computed by the system. As such, we consider it to be our contextual result (so, $Res = \{b, c, g, i, l, p\}$). The system then presents this outcome to its users which are left to judge what they think of it, using the internal meaning of the arguments contained within it, which we abstract for the sake of the example. This process is illustrated by Figure 3.48.

Some of the users are somewhat surprised by this outcome. Consider the user Arnold. Arnold remarks that the outcome is composed of propositions made by *both candidates*, whereas he expected the outcome to hold positions expressed by only one of them. He then wonders "How does this constitute a reasonable outcome?". Arnold asks here what makes *Res* a valid outcome. So, if we reformulate, he in fact asks question Q_{σ}^{Ext} for *Res* on $\mathcal{A}$ with $\sigma = Sta$.

The system then seeks to provide an explanation to Arnold. This will be an explanation $Expl_{Sta}(Res)$. Such an explanation can be seen on Figures 3.44 and 3.45. On Figure 3.44, Arnold can see that no arrow is present. As such, the result that was presented is indeed coherent. In our setting, this might mean that, contrarily to what Arnold expected, the propositions of both candidates were not necessarily in contraction with one another. On Figure 3.45, Arnold can see that no red node is isolated. As such, the result that was presented indeed attacks all the other arguments. In our setting, this might mean the point of view used as outcome corresponds to a radical one, as we wanted. This process is illustrated by Figure 3.49.

At this point, Arnold is still not convinced. He sees why the outcome that was presented is valid but believes that another one might be better. He remarks that argument p is the only proposition from Candidate 1 in the outcome, while the others have all been stated by Candidate 2. He thus wonders why is p in the outcome instead of n for instance, which is a proposition from Candidate 2 as well. By reformulating, Arnold is in fact asking question QCP_{σ}^{Mem} on $\mathcal{A}$ with $S = \{p\}$, $S' = \{n\}$, and $\sigma = Sta$.

The system then seeks to provide an explanation to Arnold. This will be an explanation $Expl_{Sta}((Res \setminus \{p\}) \cup \{n\})$. Such an explanation can be seen on Figures 3.46 and 3.47. On Figure 3.46, Arnold can see that there is an arrow from from i to n and from n to l. As such, the result in which p is replaced by n is not coherent. In our setting, this might mean that, contrarily to what Arnold thought, Candidate 2 has not been consistent in his propositions, and contradicted what he previously stated at some point. On Figure 3.47, Arnold can see that the red node p is isolated. As such, the result in which p is replaced by n does not attack all the other arguments. In our setting, this might mean that, contrarily to what Arnold thought, the propositions of Candidate 2 only are not sufficient to constitute a radical point of view like we want. This process is illustrated by Figure 3.50.

At this point, Arnold is convinced. He realizes the qualities of the result that was presented to him and acknowledges that it corresponds to the kind of result that was expected. He also recognizes that what he thought was a better result is in fact not viable for several reasons. This concludes our example.

Candidate 1	$a, d, e, f, h, j, k, m, p$
Candidate 2	b, c, g, i, l, n, o

Table 3.4: The propositions of each candidate in the debate

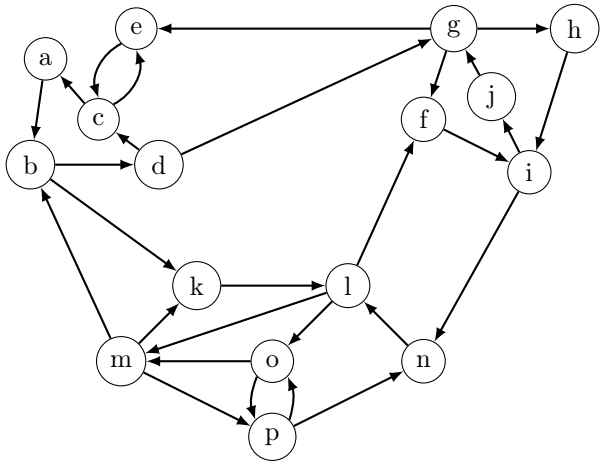

Figure 3.42: An Argumentation Framework representing a fictional debate

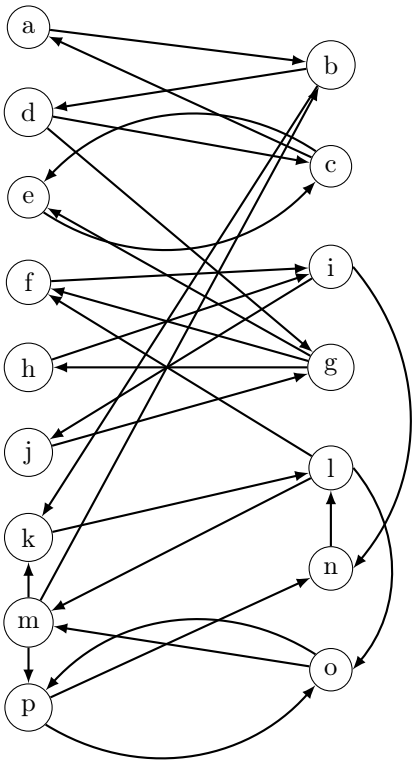

Figure 3.43: The Argumentation Framework of Figure 3.42 where the propositions of Candidate 1 are grouped together on the left and those of Candidate 2 are grouped together on the right

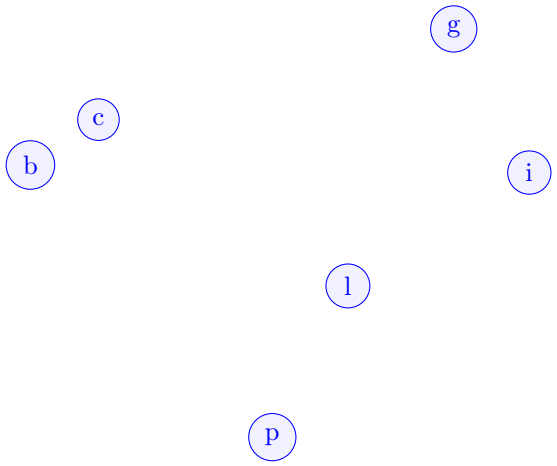

Figure 3.44: An explanation for why the outcome $\{b, c, g, i, l, p\}$ is coherent: no arrow is present

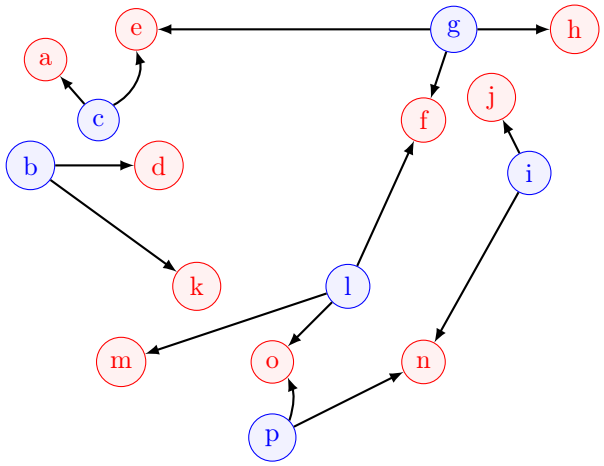

Figure 3.45: An explanation for why the outcome $\{b, c, g, i, l, p\}$ is a radical point of view: no red node is isolated

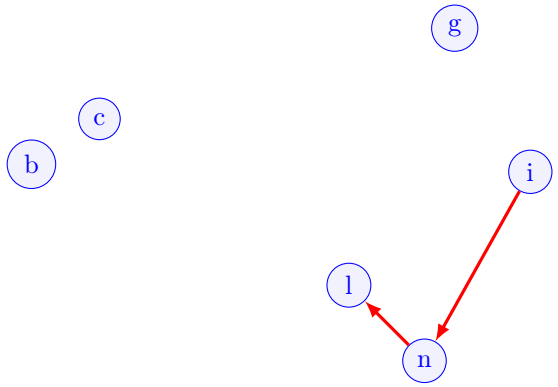

Figure 3.46: An explanation for why the outcome $\{b, c, g, i, l, p\}$, in which p is replaced by n, is not coherent: there is an arrow from i to n and from n to l

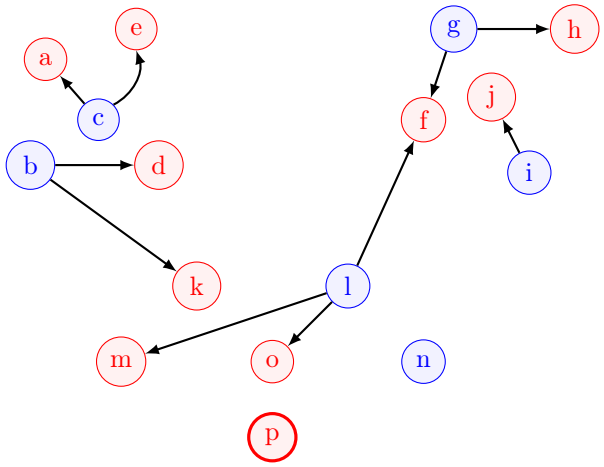

Figure 3.47: An explanation for why the outcome $\{b, c, g, i, l, p\}$, in which p is replaced by n, is not a radical point of view: the red node p is isolated

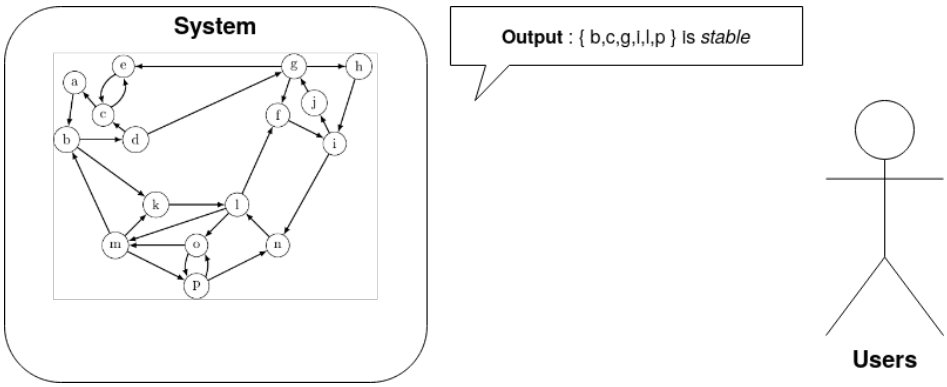

Figure 3.48: A stable extension, representing the outcome of the debate is computed and presented to the users

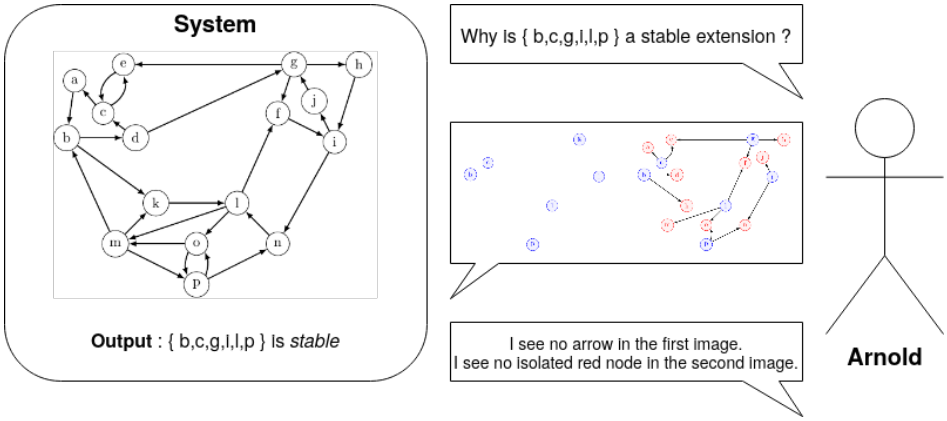

Figure 3.49: A user asks a question challenging the result as a whole, and the system provides explanations for it

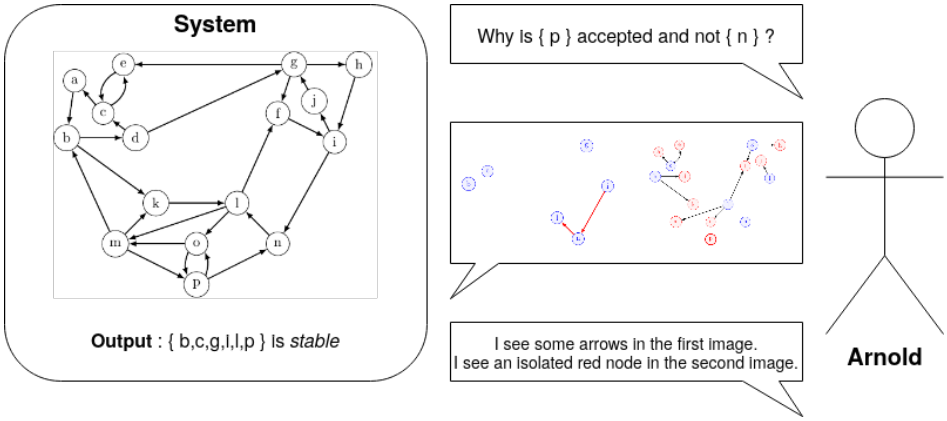

Figure 3.50: A user asks a question challenging a part of the result only, and the system provides explanations for it

3.7 Comparison with Related Works

In this section, we compare our work with those that were mentioned in Section 3.1. We will follow the same categorization from [ČRA+21].

General remarks. Before considering each related work individually, we begin with general remarks concerning our approach which make it close to, or differ from, all the other works that are mentioned. First, our work obviously falls into the category of *subgraphs*, since our explanations are subgraphs of the initial Argumentation Frameworks, for reasons that we motivated in Section 3.4.

Then, we wish to insist on the fact that we defined our explanations as *answers to precise questions*. To the best of our knowledge, we are the only ones to proceed this way. As we presented in Section 3.2.2, there are several ways of requiring explanations, and each of them may require *different answers*. Hence the importance of precisely indicating which question is tackled instead of studying explanations for a property of some object(s).

Finally, a critical difference is the efforts we made to produce explanations that rely on *visual criteria* in order to make them usable *without expert knowledge*. It is worth noting that this is not only due to the graphic nature of our explanations, but also to how they are used. All the other works mentioned define explanations that are destined for experts, either of Abstract Argumentation, or of the field in which it is applied.

Subgraphs. In [SWW20], the authors define Strongly Rejecting Subframeworks as explanations for the credulous non acceptance of some argument. A first difference with our approach is that, we defined explanations for the acceptability status of a set of arguments regarding a semantics, and for the inclusion or non-inclusion of some arguments in an extension. So, we are not interested in the same questions. In addition, there are some semantics that we do not consider in our work, namely the grounded and preferred semantics. It is also worth mentioning that, to the contrary of [SWW20], our subgraphs are not only induced subgraphs but also spanning subgraphs.

In [NJ20] and [UW21], the authors define explanations for the credulous non acceptance and acceptance of some argument respectively as sets of arguments or attacks. As we mentioned before, a first difference is that we are interested in a different problem. Their definition is *based on* the behavior of the induced (respectively spanning) subgraph resulting from the considered set of arguments (respectively attacks). Our work instead considers the subgraphs to be the explanations *themselves*. Moreover, the subgraphs we define are computed using both the induced subgraph and the spanning subgraph operations, while [NJ20] consider them separately and [UW21] only use induced subgraphs.

There also exist works that use graphs to explain, but not subgraphs. These works are [FT15a, RT21] and they rely on the concept of Defence Trees. While not being subgraphs technically speaking, one can easily retrieve the subgraph represented by a Defence Tree using the original Argumentation Framework. Hence one could wonder what are the connections between a subgraph used as an explanation and the subgraph implied by a specific Defence Tree. Alternatively, we could also explore the existence of specific Defence Trees inside a subgraph used as an explanation. So, there may exist some ties between the two approaches. Apart from the technical difference between the two methods used, a more fundamental one between the works of [FT15a, RT21] and ours is that we do not explain the same problem. Indeed, [FT15a, RT21] are interested in explaining the credulous acceptance of some argument under admissibility.

Changes. We turn to the works that consider changes as explanations, that is to say, [FT15b], [UB19] and [NJ20]. As noted in [NJ20], diagnoses can be seen as a kind of dual of the computation of induced and spanning subgraphs. Indeed, each diagnosis infers an induced or spanning subgraph, and conversely, each induced or spanning subgraph is computed using (the complement of) a diagnosis. As such, the links between the two approaches are very strong. One could thus wonder what are the properties of the complement of a set used to compute a certain induced or spanning subgraph, or what can be said about the induced or spanning subgraph computed from the complement of a given diagnosis. Although our view on explanations

is closely tied to theirs, the authors of these works seek to explain different problems from those we are addressing.

Extensions. We continue with the works that use sets of arguments as explanations, that is to say, [FT15a], [LvdT20], [BU21] and the works from Borg and Bex. We have already made some remarks on this kind of approaches in Section 3.2.1 that we will not repeat. Although the links between subgraph-based methods of explanation and extension-based methods are less direct than with diagnosis-based methods, there are still some that can be studied. Indeed, one could wonder what are the links between a subgraph computed by a subgraph-based method and the subgraph induced by the set computed by an extension-based method. Or, conversely, what can be said about the set that was used to compute an induced subgraph and the set computed by an extension-based method. Whether the explanation of an extension-based method is included in the explanation of a subgraph-based method can also be asked, the converse as well. Those are questions that could help explore the ties between the two methods, and which should be investigated in future work. Borg and Bex, as well as [FT15a] are focused on explaining the credulous and/or skeptical (non-)acceptance of some arguments, which is not the same problem as we consider. Note that [BB21b] provide a notion of contrastive explanations, just like we do with our explanations for extension membership. [LvdT20, BU21] however are interested in the same problem as our explanations for extension semantics. Yet, there is no obvious connection between their method and ours.

Dialogues. We finish with the works that use dialogue-games in their explanatory process, like [BGK$^+$14], [ABC17] and [SA18]. In [BGK$^+$14], dialogues are used as a way to obtain explanations which are in fact changes. Hence, in a way, dialogues *are* the explanatory process. On the contrary, in [ABC17] and [SA18], dialogues are the explanations. To understand the links between dialogues and graphical approaches, we should not forget that dialogues basically correspond to (parts of) an Argumentation Framework, but presented in a different, interactive, form. Starting from there, it could be possible that a dialogue obtained following a certain protocol corresponds to one of the explanations defined in the present work. Alternatively, we could try to see if the explanations we defined can be obtained via a dialogue dictated by a given protocol. If it is the case, this would mean more flexibility to our approach. Indeed, we could then present our explanations either as graphs relying on some structural properties, or as dialogues in a more interactive way. For the moment, such considerations are left as future research directions.

3.8 Quality of Explanations

In this section, we discuss the explanations that we defined in this chapter, and try to assess some of their advantages, limits, as well as their overall utility and quality.

The way we proceed is essentially by making some observations on the behavior of our explanations in certain cases. These observations were not made in previous sections, but may have nonetheless not escaped the careful reader. We make them here and now, because we consider they are some first steps in assessing the quality of our explanations, especially their limits and their potential improvements.

Our first observation concerns our explanations for Abstract Argumentation semantics, defined in Section 3.4. Recall that they are defined as answers to the questions Q_σ^{Ext} for a semantics σ. *Questions*, using the plural. Indeed, the notation Q_σ^{Ext} covers two questions, that are "Why is S a σ extension?" and "Why is S not a σ extension?". As such, a same class of answers is in fact defined for both questions. This might seem strange considering that we then proceeded to define different answers for each question based on the presence or absence of a negation in Section 3.5 for explanations on extension membership.

This difference in treatment is due to *what the questions are about*. In questions on extension membership, the questions are on the presence or absence of some argument(s) in the result. We can thus assume that the user expected the contrary of the statement used in the question, which we can use to provide answers. In questions on semantics, the questions are on the conformity of some set to the conditions of a semantics. As before, the presence or absence of a negation might indicate what the user expected (either that the set

respects the conditions or not). However, in this case, it in fact does not have any importance. Indeed, a set of arguments being an extension of some semantics is *entirely determined* by the Argumentation Framework that is considered. And remember that by our Hypothesis (H1), the Argumentation Framework of our context is *fixed*. As such, *no matter what the user may expect*, the conformity or inadequacy of S with the semantics σ is *already established*. And this conformity or inadequacy is precisely what is shown on our explanations. If the presence or absence of a negation is indeed indicative of what the user expected, it is then entirely possible that our explanations might *prove the user to be wrong*.

Example. Consider the Argumentation Framework of Figure 2.3. We have already seen that Figure 3.25 is an answer to the question `"Why does {b,h,j} not respect the Complement Attack principle?"`. It is however also an answer to the question `"Why does {b,h,j} respect the Complement Attack principle?"`. In the latter case, we might think that the user assumes $\{b, h, j\}$ to indeed respect the Complement Attack principle. In the Argumentation Framework of Figure 2.3, this is however not the case, and Figure 3.25 shows it.

This way of doing might be surprising to some. Indeed, by proceeding as we do, we deliberately might not comply with the expectations of the user. This is an understandable concern, in that the user could get annoyed by this behavior. It could be argued that, instead of showing the user that they are wrong, it would be best to align with the expectations of the user and that, if an explanation is not in accordance with the expectations of the user, then it is a sign that the system should seek to modify itself to better suit the user. While we do not necessarily disagree with such a way of doing, we wish to point out that the identification of such a discordance between the system and the user is certainly the first step of such an adaptive feedback mechanism. Our explanations as they are will make explicit such a discordance, without having the system directly taking measures, but instead letting the user judge the situation. After all, it is always possible that the user admits that they were wrong. If it is not the case, then a feedback mechanism can be initiated to make the system adapt itself to the user. However, such a mechanism is not our object of study in the present work.

The following observation concerns explanations for extension membership. Recall that our approach for these explanations consists in showing the user what they expected, with the objective of convincing the user that it is actually not possible and so that the original result is somewhat better. For instance, considering the question $QNCP_{\sigma}^{Mem}$ `"Why is` S `accepted?"`, we assume that the user expected S not to be part of the result, and thus display an explanation for Abstract Argumentation semantics on $Res \setminus S$ for σ, with the objective of showing that $Res \setminus S$ is not a viable extension for σ. However, it may very well happen that $Res \setminus S$ *indeed is* a viable extension for σ.

Example. Consider the Argumentation Framework of Figure 2.2 and the result $\{b, d, f, i\}$ for the complete semantics. Suppose the user asks the question `"Why is` $\{i\}$ `accepted?"`. According to Definition 46, the system will then show an explanation for completeness on $\{b, d, f, i\} \setminus \{i\} = \{b, d, f\}$, with the objective of showing that a problem arises to let the user conclude that i is necessary in the result and should be part of it. However, according to Table 2.2, $\{b, d, f\}$ is a complete extension, so no problem will arise.

We used question $QNCP_{\sigma}^{Mem}$ for the sake of example, but this observation can in fact be made on all questions on extension membership. So, what causes this behavior, and what to say about it? Indeed, some might find this behavior alarming, with reasonable arguments. This is the sign that the objective for which these explanations were defined might not *always* be reached. The reasons behind this particularity are the following: the explanations only take into account the *purely argumentative results*. In other terms, when explaining, we *only focus on the Argumentation Framework that was used*. If the argumentative decision is part of a larger process, this might add some additional constraints on the selection of arguments, like for instance the absolute necessary presence of a particular argument in the extension. If this is the case, and if they are relevant, then such *external constraints* should be invoked as explanations in this situation. If not, then the answer given by the system is in fact the sign that the intuition of the user was correct: *the result modified according to the expectation of the user is indeed a viable decision and could be used as a result instead of the one presented previously.*

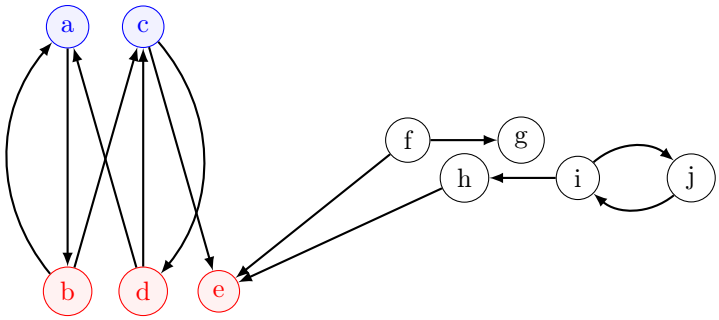

Figure 3.51: Another way to see that $\{a, c\}$ respects the Defence Principle in the AF of Figure 2.2. $\{a, c\}$ is in blue, the arguments attacked by $\{a, c\}$ are in red, and the other arguments are in white. We see that there is no arrow from a white argument to a blue argument.

Moreover, since the new potential result is built based on the expectations of the user, chances are that the user finds it better. Thus, just like for the previous observation, we believe that this situation could be used as the initiation of a feedback mechanism destined to refine the initial result presented to make it more aligned with the preferences of the user. As such, we do not really consider this behavior of our explanations to be a weakness, but rather an opening towards their personalization. If this point of view still appears unsatisfying to the reader, there is of course always the possibility of using the other approach mentioned at the beginning of Section 3.5[12] when ours fails to be considered enough.

Our next observation is on our explanations' level of transparency. Recall that, by definition, our explanations are subgraphs. As such, there are parts of the initial graph that are left aside. Moreover, in virtue of the inclusion relation between explanations on a same set of arguments for a same semantics, there is even the possibility of making additional pruning inside a given explanation (until a minimal explanation is obtained). The reason why we proceed this way is to achieve simplicity. Indeed, Theorems 1 to 5 show that our explanations can be used effectively without relying on the parts of the initial graph that are not in the explanation. Thus, we do not overwhelm the user with information that can be deemed unnecessary or irrelevant, ensuring a simpler explanation and, hopefully, a better understanding.

However, from the perspective of the user, these parts of the initial graph that are pruned away correspond to information that is *hidden*. While we argued that it makes the explanations simpler and more understandable, it could also be argued that some users might find it unsettling. To these users, maybe the hidden information was important, or maybe they just dislike by principle hiding information and thus find it suspicious. In any case, it seems that our approach is probably not the best suited for these users.

Yet, it might be possible to find different types of explanations, better suited for such users, that are still similar in nature (i.e. that are based on the same ideas). We can think of an example regarding the Defence Principle. We could define an explanation for Defence for a set of arguments on an Argumentation Framework as the entirety of the Argumentation Framework (so that no information is missing) but visually organized to highlight three partitions of the set of arguments:

1. the set of arguments the explanation is about
2. the set of arguments that is attacked by the first set
3. the other arguments

Then, the set number 1 of arguments would respect the Defence principle if and only if there is no attack from the set number 3 to the set number 1.

[12] We are talking about the approach consisting of answering question $QNCP_{\sigma}^{Mem}$ on S by showing that S "has the right to be accepted", and all the answers that may derive from this take for the other questions.

Example. Consider the Argumentation Framework of Figure 2.2. According to Table 2.2, $\{a, c\}$ is an admissible extension, and thus respects the Defence Principle. An alternative way from our explanations to see that could be by looking at Figure 3.51. It is merely a different display of the Argumentation Framework of Figure 2.2, in which the arguments have changed positions, $\{a, c\}$ is displayed in blue, the arguments attacked by $\{a, c\}$ are displayed in red, and the other arguments are displayed in white. Notice, that all the information of the initial Argumentation Framework is kept. We can conclude that $\{a, c\}$ respects the Defence Principle by seeing that there is no attack from a white argument to a or c.

It remains to be seen if similar intuitions can be found for the other semantics principles. If it is the case, the two approaches could be considered to be complementary. In the end, the choice of using one approach or the other would be a question on finding balance between two considerations: either to not bother the user by hiding information, or to not bother the user by showing irrelevant information. Of course in the present work, we present results leaning towards the latter, but approaches leaning towards the former are definitely worth investigating.

The last observation we make concerns the explanatory process, and more precisely the contextual information that we take for granted throughout this chapter. Recall from Section 3.2.2 that it is an Argumentation Framework, a semantics to compute an extension, and an actual extension taken as the result of the argumentative process. Specifically, the result is only one set of arguments. However, looking back at Section 2.1, we see that for a single Argumentation Framework, there can be potentially several extensions for the same semantics. As such, for the system to present a single extension as the result, there must have been some selection step, outside the argumentative process itself, that guided the choice of this particular result. For now, this selection step is not taken into account in our explanatory process, precisely because it is not part of the argumentative process. Yet, as it is part of the selection of the result, one could wonder if it should be part of the explanation that the user receives.

We consider that, ideally, it should indeed be part of the explanation. We did not include it in the present work because we wanted to focus specifically on the argumentative process, which, as we said, the selection of an extension among several possible ones is not a part of. For this reason, we also believe that this does not constitute a methodological mistake, or, put differently, that taking the selection of a single extension into account in the explanation invalidates or drastically changes the present work. Instead, we think that taking the selection of the result into account merely is complementing our explanations with an additional explanation for this selection, which, again, can be completely unrelated to the argumentative process.

For instance, the selection of the result could be as trivial as returning the first extension computed, or returning one at random. This certainly corresponds to situations where no specific extension holds particular importance and thus where this selection step does not really matter. In this case, the additional explanation would surely be quite simple. Alternatively, the selection step could be more elaborate, with constraints on some arguments needing to be in the extension, while others must not be part of it, and additional constraints on the size of the extension, enforcing that the extension does not have too much arguments in it. In this case, the additional explanation would be built around all these constraints. With this, we see that the selection of the result among all the possibilities should probably have its own explanatory system, which would certainly be fairly simple, but still distinct from the explanatory system of the argumentative process. Then, the explanations for the selection of the result could be used to supplement our own explanations on how this result was argumentatively obtained.

3.9 Future Perspectives

The work presented in this chapter opens many possibilities. The one we think to be the most obvious is on extending our explanations to cover the missing classical semantics, that are preferred and grounded. Following the methodology we established and the decomposition of semantics into principles, this would in fact imply the definition of explanations for the maximality and minimality of extensions. Note that this is completely independent from the notion of maximal and minimal explanations, if explanations for maximality and minimality can be defined, we can perfectly think of minimal explanations for maximality

or maximal explanations for minimality. The thing is, for such explanations to be defined following our methodology, we would need a (preferably simple) visual (i.e. structural) property that could serve as a witness of these principles on some class of subgraphs. As for now, we have not found such properties and classes of subgraphs and we do not really have an intuition to lead us on this way. One could believe that such properties and objects do not exist for maximality and minimality. This is because, recalling Table 2.4, the semantics for which we defined explanations have their Verification Problem (so the problem explained by our explanations) in L, while grounded and preferred semantics (for which we lack minimality and maximality respectively) have their Verification Problem P-complete and coNP-complete respectively. This might thus disqualify them from being explained using the same tools as the other semantics (following our methodology at least). But still, this is probably not something we can be sure of either, so in our opinion, the problem is still open.

Since the use of visual properties of graphs is at the core of our approach, it could be beneficial to investigate additional properties from Graph Theory dealing with the visual organization of graphs for our explanations. As for now, we mainly use the properties of a node being either a source node or an isolated node among a meaningful partition of a graph. We also have bipartition results for some explanations, which give additional insights as to how the graph can be visually displayed. Other instances of such visual friendly properties could include planarity for example. While the argumentative meaning of a planar graph could be unclear or irrelevant, it may be interesting to know in which situations we can guarantee an explanation to be planar because it would then be more visually appealing. We should explore Graph Theory to search for more such visual properties and see whether they could be meaningful for our explanations. If it is the case, we should then try to see in which circumstances we can provide guarantees for our explanations regarding these properties.

Another line of research would be to extend our methodology to explain more diverse subjects. For now, our explanations only deal with explaining why a result as a whole, or part of it, is valid, with the possibility of considering a contrast on the part of the result in the latter case. Now recall that our explanations are defined as answers to questions. We think that one way to extending them would be to increase the range of questions that can be answered following our methodology. For instance, consider two Argumentation Frameworks, $\mathcal{A} = (\mathscr{A}, \mathscr{R})$, $\mathcal{A}' = (\mathscr{A}', \mathscr{R}')$, a set of arguments S such that $S \subseteq \mathscr{A}$ and $S \subseteq \mathscr{A}'$ and the following question: "Why is S an admissible extension in $\mathcal{A}$ and not in $\mathcal{A}'$?". This question is of course not covered by our explanations, but might still prove relevant. What we observe is that this question is structurally very similar to the ones addressed in the present work: it uses similar contextual information we could consider that the additional Argumentation Framework is brought by the user), it deals with the validity of a set of argument regarding a semantics, and there is contrast that is made. However, this contrast is not made on another set of arguments but on another Argumentation Frameworks. It thus requires *different elements of answers*, that is different subgraphs from the ones we defined and that would show why S is supposedly valid in the first framework but not in the other. What we believe is that it might be possible to compute these elements of answers based on the question that is asked. We could imagine a formal grammar (see [Cho63] and [CM63] for basic notions on the subject) that would yield questions that are different but structurally similar (since originating from the same grammar). Then, we could imagine that the sequence of production rules used to yield a specific question could correspond to a composition of operations to make on related Argumentation Frameworks to obtain the answer. Thus, the answer would be somewhat computed *depending on* the question that is asked. To stay on the limited scope of Abstract Argumentation, we could imagine questions about Argumentation Frameworks, sets of arguments and semantics, on whether a set is valid regarding a semantics or not, and the possibility to have a contrast on either of those elements. Thus, a question like "Why is S an admissible extension in $\mathcal{A}$ and not a stable extension?" could be another different possibility captured by this way of doing. We think this line of research is very interesting, but probably also very complex and only for long term considerations. [BDDL22] provides some preliminary investigations as to how this might be achieved.

On a similar line of research, we could investigate explanations for other properties than semantical ones. Indeed, for now, our explanations only dealt with the property of being valid regarding a semantics for a set of

arguments. We could however imagine questions that are not about *semantical* but instead *structural* ones. For instance, "Why is a an argument?" or "Why is there an attack from a to b?". These are questions about how the Abstract Argumentation Framework is structured, how it is built. In the case of Abstract Argumentation, such questions are supposedly irrelevant because we consider the Argumentation Framework to be given, we do not know, nor do we care, how it was obtained. However, for a neophyte user, this might still be questions worth of interest. Again, since in Abstract Argumentation, the framework is considered to be given, the elements of answers for those questions will probably not be from the tools used in this domain. Instead, they will probably be from a domain whose main concern is to *build an Argumentation Framework* from something else. In this category could fall for instance Structured Argumentation or Argument Mining.

To keep on the track of extending our approach of explanations for Abstract Argumentation, we could think of generalising our explanations to generalised accounts of argumentation. That is to say, define our explanations for enriched frameworks. We presented a few possibilities of enrichments in Section 2.3, but many more exist which we did not cover, like for instance Claim-based Argumentation Frameworks (CAFs). In CAFs, arguments are paired with claim that they support, with the possibility of several arguments supporting the same claim. Thus, arguments are no longer abstract as the claim they support is a glimpse of their internal structure. The reader can see [DRW23] for a more detailed study of this enrichment. Since enrichments aim at capturing additional ways of arguing, it could be of interest to see how our explanations adapt to these settings, and how the addition of enrichments actually impacts the reasons behind decisions taken in more complex forms of argumentation. Indeed, recall from Section 2.3 that the addition of enrichments changes how semantics works. This would undoubtedly affect how our explanations are computed since they are precisely based on the decomposition of semantics. As such changes are profound by nature, it is not obvious to see how much our explanations would have to be adapted to a generalised account of argumentation. Thus, we do not have yet a clear intuition on how to generalise our explanations. It should be pointed out that as these enrichments usually have several different interpretations, it seems reasonable that the generalised explanations would then be for a given interpretation of some enrichment. Nonetheless, working towards explanations for a generalised account of argumentation could bring useful insights on the core mechanisms in the argumentative process.

We could also push further the use of subgraphs to explain by imagining new kinds of explanations. Consider that most semantical properties of Abstract Argumentation, and especially the fundamental property of acceptability, are purely decided by the attack relation. In the case of acceptability for instance, once the Argumentation Framework, the set of arguments and the argument, whose acceptability status with respect to the set is checked, are known, the argument being acceptable with respect to set or not is entirely dictated by the attack relation. We must indeed check that all the attackers of the arguments (those are defined by the attack relation) are attacked back by the set (again, this is only dependent on the attack relation). So the point is that, *usually*, semantical properties only rely on how the attack relation is structured (this is not always the case, for instance in the case of the grounded and preferred semantics where a property of minimality or maximality of the set is added). However, the subgraph inclusion relation does not only compare the organization of the attack relation, but also if nodes with the same names are indeed the same nodes (meaning they are connected in the same way to the same other nodes in both graphs). There exists however a relation between graphs that, contrarily to subgraph inclusion, solely compares the structure of the arcs: subgraph isomorphism. Subgraph isomorphism can be thought as a more general (and more complex) comparison then subgraph inclusion. Where subgraph inclusion more or less informally asks "Do my graphs contain the same elements?", subgraph isomorphism informally asks "Are my graphs organized in the same way?". As such, we believe that subgraph isomorphism can be used to build different kinds of explanations for Abstract Argumentation results than the ones built in this chapter. For instance, imagine that a user does not agree with a part of a result obtained via Abstract Argumentation and asks for explanations. One way to explain it could be of the likes of "The part of the graph in which you disagree with the result is isomorphic to this other part of the graph with which you agree and in which the same result is obtained. So you should agree with the first part.". This would be somewhat akin to a reasoning by association.

Finally, as a last idea, we should not forget that the explanations defined and investigated in this chapter

are designed for potentially anyone, and in particular, people without muck background in computer science. Although there are some formal results which tend to support this feature (notably Theorems 1 to 5 that show how to use and understand the explanations using only their visual characteristics), these intuitions still need confirmation from an empirical point of view. This means conducting social experiments to assess whether people that are non experts can indeed use and understand the explanations easily and whether they find them useful or not.

Chapter 4

Logical Encoding of Argumentation Frameworks to Compute Extensions

In this chapter, we present a logical encoding whose aim is to capture Abstract Argumentation Frameworks and their classical semantics. By "Abstract Argumentation Frameworks", we mean any framework that can be obtained with a combination of the enrichments presented in Section 2.3. This covers the basic Argumentation Frameworks, but also Argumentation Frameworks with Coalitions, Higher-Order Argumentation Frameworks and so on. More specifically, the first objective is to provide what we call a *generic* theory for Argumentation Frameworks and their enriched versions: a group of general formulas that are common to every framework, and in which the variable parts are isolated into what we call *parameters*. The point is to specify the particularities of each individual kind of Abstract Argumentation Framework in the parameters, which are then integrated into the shared general formulas to obtain the theory relative to the kind of Abstract Argumentation Framework considered. The second objective is that the theories obtained this way can be used to compute the extensions of the Abstract Argumentation Framework which is encoded for the classical semantics through their models.

The chapter is organized as follows: we begin with an overview of existing similar approaches (Section 4.1), the presentation of our motivations (Section 4.2) and a succinct description of the technical tool used in this chapter (Section 4.3). Next, in Section 4.4, we present an Abstract Argumentation Framework that regroupes all the enrichments that we consider in the present work and will act as the backbone of our logical encoding. Following this, we present our generic logical theory in Section 4.5, starting with the shared formulas and then showing how they can be parameterized to retrieve all the Abstract Argumentation Frameworks captured by our general formalism presented previously. Additionally, we provide results stating that the family of logical theories obtained can indeed be used to compute the extensions of the encoded Abstract Argumentation Frameworks for the classical semantics. Lastly, we summarize our work, compare it to relevant related works (Section 4.7) and discuss perspectives for future research (Section 4.8).

4.1 Existing Approaches

The links between Abstract Argumentation and Formal Logic have been studied since the introduction of the former. Indeed, already in the seminal work of Dung ([Dun95]), a first bridge between these domains has been presented, linking Abstract Argumentation and Logic Programming. The nature and objectives of these bridges are variable, although for the latter, it is often a matter of having a way to compute extensions for some Abstract Argumentation Framework (as is our case in the present work). One way to do so is through the so-called *model checking* approach.

The idea is to compute all the extensions of an Abstract Argumentation Framework for a given semantics, using a group of logical formulas. Once such a group of formulas is specified, the aim is to establish a link between its models and the extensions of the Abstract Argumentation Framework that is considered

(usually, one model corresponds to one extension). Several works have been done in this line of research (see for instance [BD04, CG09, DJWW12, CDGV13]) in the case of simple Argumentation Frameworks. More recently, a model checking approach has been developed for Higher-Order Argumentation Frameworks (and the simpler Argumentation Frameworks as well) in [CL18]. This work was then extended to Higher-Order Evidence-Based Argumentation Frameworks in [CL20] for the specific case where no support cycles are present at first, and then in the general case in [Lag21], in which even the case of coalitions of arguments is handled.

4.2 Motivation

There is, in the existing model checking approaches for the computation of extensions, an inconvenience for which the will of fixation serves as the basis of our motivation. Indeed, there is already a certain number of works proposing logical encoding of Argumentation Frameworks following the model checking approach, and even of enriched Abstract Argumentation Frameworks. However, all of these propositions are *ad hoc* constructions, dedicated to a unique kind of Abstract Argumentation Framework, without trying to rely on the similarities that connect all these frameworks.

This is something to which we want to bring an answer. The key idea of this work is *unification*. We want to propose a single *generic* logical theory that is sufficiently large to be ultimately adapted to any kind of Abstract Argumentation Framework, enriched or not. To do so, it is essential to rely extensively on the common ground of all these different frameworks. This is indeed a central point of our general methodology, as the core of our generic logical theory is in fact the group of formulas that are shared by all the Abstract Argumentation Frameworks that are included. As such, the variable parts that correspond to the individual specificities of each kind of Abstract Argumentation Framework are in fact isolated into what we call *parameters*. So, to retrieve a particular kind of Abstract Argumentation Framework, it suffices to give the correct meaning to each parameter. Since the parameters are already included in the general group of formulas (but without any meaning at first), their effect once they have been specified is immediate. Thus, the only to do to instantiate a particular kind of Abstract Argumentation Framework is indeed only to instantiate the parameters.

Such an approach is not *modular* per se. For a modular approach, one would expect a general theory that is already ready to be used for some case, and then additional formulas to add or not whether we want to enable or disable some enrichments. However, we wish to insist that the formulas that instantiate the parameters are not arbitrary. In fact, we will observe that, depending on the presence or absence of a given enrichment, these formulas are modified in the same way. Thus, the formulas corresponding to the presence of two enrichments are in fact those obtained by mechanically applying the modifications that correspond to both of them individually. As such, they are not *ad hoc* constructions.

We would also like to point out that each theory obtained through the instantiating of our generic theory is essentially the same as the logical encoding already given in the literature for the same kind of Abstract Argumentation Frameworks (in particular those described in [CL18, CL20]).

4.3 Technical Tool: First-Order Logic

We recall here some notions of First-Order Logic that we will use in this chapter. We refer the reader to [Hod13] (of which most of the definitions and results presented in this section are from) for additional notions on this subject.

We suppose the reader familiar with the notion of Propositional Logic, of which First-Order Logic is an extension. As for any formal logic, the point of First-Order Logic is to model *statements* and then have a way of deciding which statements are true and which are false. As such, the first step is to define an abstract language using which we can write syntactical expressions that aim at capturing the statements we wish to model, independently from what these statements actually are. There are two kinds of symbols used in a language of First-Order Logic, given in Table 4.1.

Logical symbols		Nonlogical symbols	
$x, y, z, \dots$	individual variables	$a, b, c, \dots$	constant symbols
$\neg \vee$	logical connectors	$f, g, h, \dots$	n-ary function symbols ($n = 1, 2, \dots$)
$\forall$	quantifier	$P, Q, R, \dots$	n-ary predicate symbols ($n = 1, 2, \dots$)
$(\ )\ ,$	punctuation		
$=$	symbol for equality		

Table 4.1: Symbols of the language of First Order Logic

Note. We assume that there is an infinite list of individual variables, but that each kind of nonlogical symbols has a countable number of elements.

Note. In practice, the constant symbols, function symbols and predicate symbols are specific to *a given language*, and precisely serve to differentiate a language from another. When defining a specific language, those symbols must be specified. The ones given in Table 4.1 can be thought as *metasymbols*: symbols used to denote any symbol that can be put in their place.

Convention. We introduce additional logical symbols to the language that are defined relatively to those given in Table 4.1. $(A \wedge B)$ means $\neg(\neg A \vee \neg B)$, $(A \to B)$ means $(\neg A \vee B)$, $(A \leftrightarrow B)$ means $((A \to B) \wedge (B \to A))$, and $\exists x_n(A)$ means $\neg\forall x_n(\neg A)$.

The syntactical expressions used to represent the statements we wish to capture are called *formulas*. They are defined using the symbols given in Table 4.1. They rely on an intermediate syntactical concepts, the *terms*.

Definition 52. The *terms* are defined inductively as follows:

1. Each individual variable x and each constant c is a term.
2. If f is an n-ary function symbol and $t_1, \dots, t_n$ are terms, then $f(t_1, \dots, t_n)$ is a term.
3. Every term is obtained by a finite application of steps 1 and 2.

Terms can be thought as representing individual objects our statements are about. Using the terms, we can define the formulas.

Definition 53. The *formulas* are defined inductively as follows:

1. If s and t are terms, $(s = t)$ is a formula.
2. If P is an n-ary predicate symbol and $t_1, \dots, t_n$ are terms, then $P(t_1, \dots, t_n)$ is a formula.
3. If A and B are formulas and x_n is an individual variable, then $\neg A$, $A \vee B$, and $\forall x_n(A)$ are formulas
4. Every formula is obtained by a finite application of steps 1, 2 and 3.

The formulas can be thought as the "correct" syntactical expressions of the language. In particular, they are the expressions we are interested into, as they are the ones that are meant to represent the statements we wish to model. Thus, we now need a way to determine whether a formula (and so the statement it represents) is true or not. To this end, we define the concept of *interpretation*.

Definition 54. An *interpretation* of a first-order language consists of:

1. a *non-empty* set D_I (called the domain of the interpretation);
2. for each constant symbol c, a specific element c_I of D_I (so $c_I \in D_I$);
3. for each n-ary function symbol f, an n-ary operation f_I on D_I (so $f_I : D_I^n \mapsto D_I$)

4. for each n-ary predicate symbol P, an n-ary relation P_I on D_I (so $P_I \subseteq D_I^n$).

An interpretation is a well named object, in that it is indeed intended to model the meaning we give to the nonlogical symbols of a language (so that its formulas indeed represent the statements we wish to model). To know whether a formula is true or not, we also need the concept of *assignment* in an interpretation.

Definition 55. Let I be an interpretation of a first-order language. An *assignment in* I is a function ϕ from the set of individual variables into the domain of I.

Note. If we denote by VAR the set of individual variables of a first-order language, an assignment is thus a function ϕ such that $\phi : VAR \mapsto D_I$.

An assignment can be thought as a way of giving a precise meaning (one among possibly many) to a formula regarding an interpretation. We now introduce a notion used to determine if two assignments are somewhat "close" to each other.

Definition 56. Let I be an interpretation of a first-order language. Two assignments ϕ and ψ in I are *x_n-variants* if $\phi(x_k) = \psi(x_k)$ for all $k \neq n$.

Note. Two assignments ϕ and ψ of an interpretation I that are x_n-variants in fact agree everywhere except possibly at x_n where they may or may not differ.

Remark. Every assignment ϕ of an interpretation I is a x_n-variant of itself.

To determine whether a formula is true or not, we first need to extend the notion of assignment to terms, and then to formulas. This is the object of the two following definitions.

Definition 57. Let I be an interpretation of a first-order language and ϕ an interpretation in I. The extension of ϕ to terms, which we continue to denote ϕ, is defined by induction as follows:

1. For each constant symbol c of the language, $\phi(c) = c_I$.
2. If f is an n-ary function symbol and $t_1, \ldots, t_n$ are terms, then $\phi(f(t_1, \ldots, t_n)) = f_I(\phi(t_1), \ldots, \phi(t_n))$.

Note. If we denote by TRM the set of terms of a first-order language, the extension of an assignment to terms is thus a function ϕ such that $\phi : TRM \mapsto D_I$.

Definition 58. Let I be an interpretation of a first-order language and ϕ an interpretation in I. The extension of ϕ to formulas, which we continue to denote ϕ, is defined by induction as follows:

1. If s and t are terms, $\phi(s = t) = true$ if $\phi(s) = \phi(t)$ (so $\phi(s)$ and $\phi(t)$ are the same element of D_I) and $\phi(s = t) = false$ otherwise.
2. If P is an n-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $\phi(P(t_1, \ldots, t_n)) = true$ if $(\phi(t_1), \ldots, \phi(t_n)) \in P_I$, and $\phi(P(t_1, \ldots, t_n)) = false$ otherwise.
3. If A is a formula, then $\phi(\neg A) = true$ if $\phi(A) = false$, and $\phi(\neg A) = false$ otherwise.
4. If A and B are formulas, then $\phi(A \vee B) = true$ if $\phi(A) = true$ or $\phi(B) = true$, and $\phi(A \vee B) = false$ otherwise.
5. If A is a formula and x_n is an individual variable, then $\phi(\forall x_n(A)) = true$ if $\phi(A) = true$ for every assignment ψ that is an x_n-variant of ϕ, and $\phi(\forall x_n(A)) = false$ otherwise.

Note. If we denote by FRM the set of formulas of a first-order language, the extension of an assignment to formulas is thus a function ϕ such that $\phi : FRM \mapsto \{true, false\}$.

We can now advance towards the definition of what is a true formula of a first-order language.

Vocabulary. Let I be an interpretation of a first-order language, A a formula over that first-order language and ϕ an assignment in I. We say that ϕ *satisfies* A if $\phi(A) = true$.

Vocabulary. Let A be a formula over a first-order language. We say that A is *satisfiable* if there is at least one interpretation I of that first-order language and at least one assignment ϕ in I such that ϕ satisfies I. If A is not satisfiable, we then say that it is *unsatisfiable*.

Definition 59. Let I be an interpretation of a first-order language and A a formula over that first-order language. A is *true in the interpretation* I if every assignment of I satisfies A.

Remark. A formula A is not defined as being true *per se*, but instead is defined as being true *in a given interpretation*.

Vocabulary. Let I be an interpretation of a first-order language and A a formula over that first-order language. We say that I is a *model* of A if A is true in I. This extends to sets of formulas: if Γ is a set of formulas over the language of I, we say that I is a model of Γ if I is a model of every formula of Γ.

Vocabulary. Let I be an interpretation of a first-order language and A a formula over that first-order language. We say that A is *false in the interpretation* I if no assignment in I satisfies A.

Definition 60. Let A be a formula over a first-order language. A is *logically valid* if it is true in every interpretation I of the language of A.

Although determining whether a given formula (or set of formulas) is logically valid or even satisfiable in Propositional Logic is decidable, it is well known that the same problem is *undecidable* in general in First-Order Logic. This means that there exists no algorithm that can tell whether an arbitrary formula (or set of formulas) over a first-order language is satisfiable or valid. To be more precise, this problem is in fact *semidecidable*: we have an algorithm that can enumerate all logically valid formulas over the language (so we might find our formula(s) among them if we wait long enough), but that is all. In particular, if our formula (or one of our formulas) is not logically valid, the algorithm never stops. We also know that there exists particular restricted fragments of first-order logic that are decidable.

In practice, we are sometimes not interested in verifying whether a formula is satisfiable *in general* (so considering every possible interpretation), but rather whether *one specific interpretation* satisfies it. So it is more a problem of determining if a certain interpretation satisfies our formula rather than determining whether there exists an interpretation that satisfies it. This is the problem that we are interested in in this chapter. Note that this problem, provided that the interpretation has a *finite domain* (this is not necessarily the case according to Definition 54) is indeed decidable.

We now characterize a specific kind of interpretation, designed to be as simple as possible: Herbrand models. These will be the models that we will use in this chapter. For this, we need the notions of Herbrand universe and Herbrand base.

Definition 61. Let A be a formula over a first-order language. We define:

1. $\mathfrak{U}_0$ is the set of all constant symbols in A. If A has no constant symbol, $\mathfrak{U}_0$ is $\{a\}$ instead (so that it is not empty)
2. $\mathfrak{U}_{i+1}$ is the union of $\mathfrak{U}_i$ with the set of all the terms of the form $f_n(t_1, \ldots, t_n)$ where f_n is an n-ary function symbol that appears in A and $t_1, \ldots, t_n$ are terms of $\mathfrak{U}_i$

The Herbrand universe of A, denoted $\mathfrak{U}_A$, is the set of terms $\bigcup_{i=0}^{\infty} \mathfrak{U}_i$.

The Herbrand universe of a formula can be thought of as the set of all terms that are relevant for this particular formula. This definition can be easily extended to sets of formulas. Now, we give the definition of the Herbrand base of a formula.

Definition 62. Let A be a formula over a first-order language. The Herbrand base of A, denoted $\mathfrak{B}_A$, is the set of all formulas of the form $(s = t)$ or of the form $P_n(t_1, \ldots, t_n)$ where P_n is an n-ary predicate symbol that appears in A and $s, t, t_1, \ldots, t_n$ are terms of the Herbrand universe $\mathfrak{U}_A$ of A.

Similarly as with the Herbrand universe of a formula, the Herbrand base of a formula can be thought as the set of all ground formulas (atoms) that are relevant for this particular formula. This definition can also be easily extended to sets of formulas.

The Herbrand base of a formula allows to easily define interpretations that are relevant for the formula we consider. Indeed, in virtue of Definition 58, the truth value of a formula derives from the truth values of the atoms that are relevant to it. This is precisely what contains the Herbrand base. So we only need to select a subset of that base, which will be the atoms that we arbitrarily consider to be true, making all the other automatically false, to obtain an interpretation. Such an interpretation is called a Herbrand interpretation.

Definition 63. Let A be a formula over a first-order language. A Herbrand interpretation is a subset of $\mathfrak{B}_A$.

This definition can also be extended to sets of formulas. We point out that Herbrand interpretations are indeed interpretations in that it is possible to build an interpretation of the first-order language (so in the sense of Definition 54) from any of them.

Vocabulary. Let A be a formula over a first-order language and I be a Herbrand interpretation. Just like arbitrary interpretations, we say that I is a *Herbrand model* of A if it satisfies A.

Notation. Let A be a formula over a first-order language. The set of Herbrand models of A is denoted $\mathfrak{H}_A$. This extends to sets of formulas.

From there on, we will assume the reader familiar with some more advanced but still classical concepts of formal logic, notably semantical entailment ($\models$) and syntactical entailment ($\vdash$). We provide additional definitions surrounding the latter.

Definition 64. Let H be a set of formulas. We say that H is *theoretically closed* if for every formula φ such that $H \vdash \varphi$, we have $\varphi \in H$.

Definition 65. Let H be a set of formulas. The *theoretical closure* of H is its smallest superset that is theoretically closed.

Notation. Let H be a set of formulas. The theoretical closure of H is denoted $\mathrm{Th}(H)$.

Finally, we give a convention that we will use for the rest of the present work when writing formulas. This convention aims at avoiding nested implicative formulas.

Convention. We will abbreviate formulas $\forall z \ \big(P(z) \rightarrow \psi\big)$ as $\forall z \in P \ \psi$ and $\exists z \ \big(P(z) \wedge \psi\big)$ as $\exists z \in P \ \psi$

Note. Please consider that this convention does not mean that we use multi-sorted logics: this is simply a convenient notation to improve the readability of the axioms further in the present work.

4.4 A General Account of Enriched Argumentation Frameworks

In this section, we are interested in defining an Abstract Argumentation Framework that captures all the enrichments mentioned in Section 2.3. Just like in Section 2.3, we will study how its semantics can be defined. This framework will be the backbone of the family of logical theories we will provide in the next section. We will also give some intuitions as to how simpler frameworks and their specific semantics can be retrieved from this one.

4.4.1 Higher-Order Bipolar Argumentation Frameworks with Coalitions

The formalism we call Higher-Order Bipolar Argumentation Frameworks with Coalitions (HO-BAF-C) is a generalisation of Argumentation Frameworks in which we consider simultaneously higher-order relations (attack and support relations) and coalitions of arguments.

Definition 66. An HO-BAF-C is a tuple $(\mathscr{A}, \mathscr{R}, \mathscr{S}, s, t)$ where $\mathscr{A}$ is a set of arguments, $\mathscr{R}$ is a set of attacks and $\mathscr{S}$ is a set of supports such that:

- $\mathscr{A} \cap \mathscr{R} = \mathscr{A} \cap \mathscr{S} = \mathscr{R} \cap \mathscr{S} = \varnothing$,
- $s : \mathscr{R} \cup \mathscr{S} \to 2^{\mathscr{A}} \setminus \{\varnothing\}$,
- $t : \mathscr{R} \cup \mathscr{S} \to \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$.

Vocabulary. We collectively refer to attacks and supports as *interactions*, and to arguments, attacks and supports as *elements*.

Vocabulary. For a Higher-Order Bipolar Argumentation Frameworks with Coalitions $(\mathscr{A}, \mathscr{R}, \mathscr{S}, s, t)$ and an interaction $\alpha \in \mathscr{R} \cup \mathscr{S}$, $s(\alpha)$ is the source of α and $t(\alpha)$ is the target of α.

The same remarks concerning the interpretations of enrichments that have been made in Section 2.3 can be made concerning an HO-BAF-C. We recall them briefly. In the case of Coalitions, we consider that an interaction is effective when all the arguments of its source satisfy a particular condition, and restrict ourselves to the case of interactions having a single target. We use the RAF interpretation of higher-order relations. Finally, we interpret the support relation as being an evidential support.

Note that from the last point, we do not just consider Higher-Order Bipolar Argumentation Frameworks with Coalitions, but more specifically Higher-Order Evidence-Based Argumentation Frameworks with Coalitions (HO-EBAF-C). In particular, this means that we must identify the origin of supports, that is to say, *prima-facie* objects. Observe that in an HO-EBAF-C, contrarily to a simple EBAF, arguments, attacks and supports can be targeted by interactions. In particular, this means that arguments, attacks and supports can all *receive* supports. Consequently, prima-facie objects, so objects whose support is not to be questioned, can be found among all the elements of the framework.

Definition 67. An HO-EBAF-C is a tuple $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ where $\mathscr{A}$ is a set of arguments, $\mathscr{R}$ is a set of attacks and $\mathscr{S}$ is a set of supports such that:

1. $\mathscr{A} \cap \mathscr{R} = \mathscr{A} \cap \mathscr{S} = \mathscr{R} \cap \mathscr{S} = \varnothing$,
2. $s : \mathscr{R} \cup \mathscr{S} \to 2^{\mathscr{A}} \setminus \{\varnothing\}$,
3. $t : \mathscr{R} \cup \mathscr{S} \to \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$,
4. $\mathscr{P} \subseteq \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$.

Vocabulary. For a Higher-Order Evidence-Based Argumentation Frameworks with Coalitions $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ and an element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$, we say that x is *prima-facie* if and only if $x \in \mathscr{P}$.

Remark. Definition 66 will not be of much use in the present work. Since we focus on the evidential interpretation of support, we will use Definition 67 instead. Nevertheless, we introduced Definition 67 through Definition 66 to follow a similar methodology as in Chapter 2 when we presented the support relation.

4.4.2 Structures and Semantics

We already discussed in Section 2.3 that in the case of Higher-Order Argumentation Frameworks, since both arguments and attacks can be targeted, the semantical concepts of Abstract Argumentation are defined relatively to a subset of arguments and a subset of attacks. Following this idea, in a Higher-Order Bipolar Argumentation Frameworks with Coalitions (and so, in a Higher-Order Evidence-Based Argumentation Frameworks with Coalitions), the semantical concepts are defined relatively to a subset of arguments, a subset of attacks and a subset of supports. Contrarily to Section 2.3, this idea is formalized by using the concept of structure, as introduced in [CFFL18a].

Definition 68. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. A *structure* is an ordered pair $U = (S, \Gamma, \Delta)$ where $S \subseteq \mathscr{A}$, $\Gamma \subseteq \mathscr{R}$ and $\Delta \subseteq \mathscr{S}$.

Remark. Please observe that, in a structure $U = (S, \Gamma, \Delta)$, S, Γ and Δ are pairwise disjoint (in symbols, $S \cap \Gamma = S \cap \Delta = \Delta \cap \Gamma = \varnothing$) as a consequence of the condition $\mathscr{A} \cap \mathscr{R} = \mathscr{A} \cap \mathscr{S} = \mathscr{R} \cap \mathscr{S} = \varnothing$ in Definition 67.

A structure is meant to capture the whole context that underlies acceptability (in a generalized account, it may take some attacks and/or supports to be accepted for a set of arguments to be accepted). Since the more elements we have to consider in definitions, the more intricate they become, we have to differ a little from the general methodology that was followed in Sections 2.1 and 2.3 to define semantics. Here, we adopt a more set-theoretic approach and define several intermediate notions before defining semantics themselves.

The first of these intermediate notions is that of elements being defeated by a structure.

Definition 69. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ a structure. An element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ is *defeated* by U if and only if there exists $\alpha \in \Gamma$ such that $s(\alpha) \subseteq S$ and $t(\alpha) = x$.

Notation. In the following, considering a structure $U = (S, \Gamma, \Delta)$, we denote by $Def(U)$ the set of all elements defeated by U.

Considering interactions, their effect can be prevented either by being defeated by a structure, or having an argument of their source being defeated by that structure. We formalize this idea with the notion of inhibition.

Definition 70. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ a structure. An interaction β is *inhibited* by U if and only if β or some $b \in s(\beta)$ is defeated by U.

Notation. In the following, considering a structure $U = (S, \Gamma, \Delta)$, we denote by $Inh(U)$ the set of all elements inhibited by U.

The next notion is the generalisation of evidential support to the context of Higher-Order Evidence-Based Argumentation Frameworks with Coalitions.

Definition 71. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ be a structure. An element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ is *e-supported* by U if and only if either $x \in \mathscr{P}$ or there exists $\alpha \in \Delta$ with

1. $t(\alpha) = x$,
2. $s(\alpha) \subseteq S$ and
3. α is *e-supported* by $U' = (S \setminus \{x\}, \Gamma \setminus \{x\}, \Delta \setminus \{x\})$ and for all $a \in s(\alpha)$, a is *e-supported* by U'.

Notation. In the following, considering a structure $U = (S, \Gamma, \Delta)$, we denote by $Supp(U)$ the set of all elements supported by U.

We now introduce a somewhat new notion, that of unsupportability. Informally, we consider an element to be unsupportable when we know it cannot receive any support whatsoever.

Definition 72. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ be a structure. An element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ is *unsupportable* by U if and only if x is *not e-supported* by the structure $U' = (\mathscr{A} \setminus Def(U), \mathscr{R}, \mathscr{S} \setminus Def(U))$.

Notation. In the following, considering a structure $U = (S, \Gamma, \Delta)$, we denote by $UnSupp(U)$ the set of all elements unsupportable by U.

With this, we now have everything we need to proceed with the definition of acceptability.

Definition 73. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ be a structure. An element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ is *acceptable* with respect to U if and only if

1. x is e-supported by U and
2. for all $\alpha \in \mathscr{R}$, if $t(\alpha) = x$ then either α or some $a \in s(\alpha)$ is either defeated by U or unsupportable by U.

Notation. In the following, considering a structure $U = (S, \Gamma, \Delta)$, we denote by $Acc(U)$ the set of all elements acceptable by U.

As usual, with acceptability comes the possibility of finally introducing the classical semantics.

Definition 74. Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. A structure $U = (S, \Gamma, \Delta)$ is said to be:

- *conflict-free* iff for all $\alpha \in \Gamma$, for all $x \in S \cup \Gamma \cup \Delta$, if $s(\alpha) \subseteq S$ then $t(\alpha) \neq x$,
- *admissible* iff U is conflict-free and for all $x \in S \cup \Gamma \cup \Delta$, x is acceptable wrt U,
- *complete* iff U is admissible and for all $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$, if x is acceptable wrt U then $x \in S \cup \Gamma \cup \Delta$,
- *preferred* iff U is a $\subseteq$-maximal admissible structure,[1]
- *grounded* iff U is a $\subseteq$-minimal complete structure,
- *stable* iff $(\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}) \setminus (S \cup \Gamma \cup \Delta) = \{x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S} \mid x$ is not e-supported wrt $(\mathscr{A} \setminus Def(U), \mathscr{R}, \mathscr{S} \setminus Def(U))$ or $\exists \alpha \in \Gamma,\ s(\alpha) \subseteq S,\ t(\alpha) = x\}$.

We can use some of our previously introduced notations to rewrite some semantics of Definition 74 as follows.

Corollary 3. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ be a structure. Consider the structure $U' = (\mathscr{A} \setminus Def(U), \mathscr{R}, \mathscr{S} \setminus Def(U))$. U is said to be:*

- admissible *iff U is conflict-free and $S \cup \Gamma \cup \Delta \subseteq Acc(U)$,*
- complete *iff U is admissible and $Acc(U) \subseteq S \cup \Gamma \cup \Delta$,*
- stable *iff*
  - $\mathscr{A} \setminus S = \{a \in \mathscr{A} \mid a \notin Acc(U')$ *or* $\exists \beta \in \Gamma$ *such that* $s(\beta) \subseteq S$ *and* $t(\beta) = a\}$,
  - $\mathscr{R} \setminus \Gamma = \{\alpha \in \mathscr{R} \mid \alpha \notin Acc(U')$ *or* $\exists \beta \in \Gamma$ *such that* $s(\beta) \subseteq S$ *and* $t(\beta) = \alpha\}$, *and*
  - $\mathscr{S} \setminus \Delta = \{\alpha \in \mathscr{S} \mid \alpha \notin Acc(U')$ *or* $\exists \beta \in \Gamma$ *such that* $s(\beta) \subseteq S$ *and* $t(\beta) = \alpha\}$.

4.4.3 From a General Formulation to its Usual Formulation

The definitions we gave in the previous section are general in the sense that they reduce to the correct simpler definitions corresponding to the case where some enrichments are not considered. We could have stopped at this statement and let the reader verify it by themselves, but in order to be convincing, we propose here a mechanical approach to retrieve simpler formulations of definitions from the general ones, based on how to get from the general framework to the simpler ones.

The key here is to see that retrieving simpler frameworks is mostly done through imposing restrictions upon the conditions for s and/or t in Definition 67.[2]. That is, restricting $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ in Condition 3 to $\mathscr{A}$ allows us to disable higher-order relations. Restricting $2^{\mathscr{A}} \setminus \{\varnothing\}$ in Condition 2 to singleton subsets of $\mathscr{A}$ allows us to disable coalitions. Finally, restricting $\mathscr{R} \cup \mathscr{S}$ in Conditions 2, 3 and 4 to $\mathscr{R}$ (both in domain and co-domain in the case of 3) allows us to disable the support relation. In the case of disabling the support

[1] We consider that $(X, Y, Z) \subseteq (P, Q, R)$ if and only if $X \subseteq P$, $Y \subseteq Q$ and $Z \subseteq R$.

[2] Obvious adjustments apply: e.g., a condition such that $t(\beta) \in s(\alpha)$ is identified with $t(\beta) = a$ whenever $s(\alpha)$ is the singleton set $\{a\}$.

relation, we must add that this case is identified with the situation $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$ in the general framework.

Based on these restrictions, *at the level of the framework*, we now propose the following language operations to apply directly on the definition to obtain the correct definitions from the general ones considering which restrictions we wish to apply.

Procedure for versions reducing structures to extensions (†)
Everywhere:

- replace "structure" with "set of arguments",
- replace $U = (S, \Gamma, \Delta)$ (or (S, Γ, Δ) or U) by S,
- replace $S \cup \Gamma \cup \Delta$ by S,
- replace Γ by $\mathscr{R}$,
- replace Δ by $\mathscr{S}$,
- replace $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ by $\mathscr{A}$,
- replace framework names of the form "HO-X" by "X".

Remark. Observe that the notion of structure simplifies to the notion of extension for a framework without higher-order relations considering that $\Gamma = \mathscr{R}$ and $\Delta = \mathscr{S}$. In this case, for any extension S of the simpler framework, the corresponding structure in the general framework is $U = (S, \mathscr{R}, \mathscr{S})$.

Simplification for versions disallowing coalitions (‡)
Everywhere:

- replace $a \in s(\gamma)$ by $a = s(\gamma)$,
- replace $s(\gamma) \subseteq S$ by $s(\gamma) \in S$,
- replace framework names of the form "X-C" by "X".

Simplification for versions disallowing the support relation (¶)
Everywhere:

- replace $\mathscr{S}$ by $\varnothing$,
- replace $\mathscr{P}$ by $\mathscr{A} \cup \mathscr{R}$,
- replace framework names of the form "EBAF" by "AF".

Procedure for eliminating names of attacks (‖)
Everywhere:

- replace $Q\gamma \in \mathscr{R}$ by $Qy_\gamma z_\gamma$ s.t. $y_\gamma \mathscr{R} z_\gamma$ (where Q is $\forall$ or $\exists$ or non-existent),
- replace $s(\gamma)$ by y_γ,
- replace $t(\gamma)$ by z_γ,
- delete conditions involving an equation of the form "$\gamma = \ldots$",
- delete conditions of the form "γ is $\ldots$".

Please observe how the language operations correspond, for each enrichment, to the restriction on functions s and t that we previously discussed. As an illustration, we can check that the original setting of Argumentation Frameworks is retrieved by applying all these operations on some definition.

Lemma 11. *For an Argumentation Framework, Definition 73 then becomes:*

> *Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff for all y_β, z_β s.t. $y_\beta \mathscr{R} z_\beta$, if $z_\beta = x$ then there exists y_γ, z_γ s.t. $y_\gamma \mathscr{R} z_\gamma$ where $y_\gamma \in S$ and $z_\gamma = b$ for some $b = y_\beta$.*

Equivalently,

> *Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff for all y s.t. $y \mathscr{R} x$, there exists v s.t. $v \mathscr{R} y$ where $v \in S$.*

which amounts to Definition 2.

Proof. Definition 73 is as follows (replacing several notions with their definition):

Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ a structure. An element $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ is acceptable wrt U iff (1.1) $x \in \mathscr{P}$ or (1.2) there exists $\alpha \in \Delta$ with (1.2.1) $t(\alpha) = x$, (1.2.2) $s(\alpha) \subseteq S$, and (1.2.3.1) α is e-supported by $U' = (S \setminus \{x\}, \Gamma \setminus \{x\}, \Delta \setminus \{x\})$ and (1.2.3.2) for all $a \in s(\alpha)$, a is e-supported by U', and (2) for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then either (2.1) there exists $\gamma \in \Gamma$ where (2.1.1) $s(\gamma) \subseteq S$ and either (2.1.2) $t(\gamma) = \beta$ or (2.1.3) $t(\gamma) = b$ for some $b \in s(\beta)$, or (2.2) $\beta \notin \mathscr{P}$ and there exists no $\delta \in \Delta$ with (2.2.1) $t(\delta) = \beta$, (2.2.2) $s(\delta) \subseteq \mathscr{A} \setminus Def(U)$, (2.2.3) δ is e-supported by $U'' = (\mathscr{A} \setminus (Def(U) \cup \{\beta\}), \mathscr{R} \setminus \{\beta\}, \mathscr{S} \setminus (Def(U) \cup \{\beta\}))$, and (2.2.4) for all $c \in s(\delta)$, c is e-supported by U'', or (2.3) for some $d \in s(\beta)$, $d \notin \mathscr{P}$ and there exists no $\delta \in \Delta$ with (2.3.1) $t(\delta) = b$, (2.3.2) $s(\delta) \subseteq \mathscr{A} \setminus Def(U)$, (2.3.3) δ is e-supported by U'', and (2.3.4) for all $c \in s(\delta)$, c is e-supported by U''.

Applying (†) then gives (modifications are given in a box):

Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an $\boxed{\text{EBAF-C}}$ and $\boxed{S}$ a $\boxed{\text{set of arguments}}$. An element $x \in \boxed{\mathscr{A}}$ is acceptable wrt $\boxed{S}$ iff (1.1) $x \in \mathscr{P}$ or (1.2) there exists $\alpha \in \boxed{\mathscr{S}}$ with (1.2.1) $t(\alpha) = x$, (1.2.2) $s(\alpha) \subseteq S$, and (1.2.3.1) α is e-supported by $\boxed{S \setminus \{x\}}$, and (1.2.3.2) for all $a \in s(\alpha)$, a is e-supported by $\boxed{S \setminus \{x\}}$, and (2) for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then either (2.1) there exists $\gamma \in \boxed{\mathscr{R}}$ where (2.1.1) $s(\gamma) \subseteq S$ and either (2.1.2) $t(\gamma) = \beta$ or (2.1.3) $t(\gamma) = b$ for some $b \in s(\beta)$, or (2.2) $\beta \notin \mathscr{P}$ and there exists no $\delta \in \boxed{\mathscr{S}}$ with (2.2.1) $t(\delta) = \beta$, (2.2.2) $s(\delta) \subseteq \mathscr{A} \setminus \boxed{Def(S)}$, (2.2.3) δ is e-supported by $\boxed{\mathscr{A} \setminus (Def(S) \cup \{\beta\})}$, and (2.2.4) for all $c \in s(\delta)$, c is e-supported by $\boxed{\mathscr{A} \setminus (Def(S) \cup \{\beta\})}$, or (2.3) for some $d \in s(\beta)$, $d \notin \mathscr{P}$ and there exists no $\delta \in \boxed{\mathscr{S}}$ with (2.3.1) $t(\delta) = b$, (2.3.2) $s(\delta) \subseteq \mathscr{A} \setminus \boxed{Def(S)}$, (2.3.3) δ is e-supported by $\boxed{\mathscr{A} \setminus (Def(S) \cup \{\beta\})}$, and (2.3.4) for all $c \in s(\delta)$, c is supported by $\boxed{\mathscr{A} \setminus (Def(S) \cup \{\beta\})}$.

Applying (‡) then gives (modifications are given in a box):

Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an $\boxed{\text{EBAF}}$ and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff (1.1) $x \in \mathscr{P}$ or (1.2) there exists $\alpha \in \mathscr{S}$ with (1.2.1) $t(\alpha) = x$, (1.2.2) $s(\alpha) \boxed{\in} S$, and (1.2.3.1) α is e-supported by $S \setminus \{x\}$, and (1.2.3.2) for all $a \boxed{=} s(\alpha)$, a is e-supported by $S \setminus \{x\}$, and (2) for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then either (2.1) there exists $\gamma \in \mathscr{R}$ where (2.1.1) $s(\gamma) \boxed{\in} S$ and either (2.1.2) $t(\gamma) = \beta$ or (2.1.3) $t(\gamma) = b$ for some $b \boxed{=} s(\beta)$, or (2.2) $\beta \notin \mathscr{P}$ and there exists no $\delta \in \mathscr{S}$ with (2.2.1) $t(\delta) = \beta$, (2.2.2) $s(\delta) \boxed{\in} \mathscr{A} \setminus Def(S)$, (2.2.3) δ is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, and (2.2.4) for all $c \boxed{=} s(\delta)$, c is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, or (2.3) for some $d \boxed{=} s(\beta)$, $d \notin \mathscr{P}$ and there exists no $\delta \in \mathscr{S}$ with (2.3.1) $t(\delta) = b$, (2.3.2) $s(\delta) \boxed{\in} \mathscr{A} \setminus Def(S)$, (2.3.3) δ is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, and (2.3.4) for all $c \boxed{=} s(\delta)$, c is supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$.

Applying (¶) then gives (modifications are given in a box):

Let $(\mathscr{A}, \mathscr{R}, \boxed{\varnothing}, \boxed{\mathscr{A} \cup \mathscr{R}}, s, t)$ be an $\boxed{\text{AF}}$ and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff (1.1) $x \in \boxed{\mathscr{A} \cup \mathscr{R}}$ or (1.2) there exists $\alpha \in \boxed{\varnothing}$ with (1.2.1) $t(\alpha) = x$, (1.2.2) $s(\alpha) \in S$, and (1.2.3.1) α is e-supported by $S \setminus \{x\}$, and (1.2.3.2) for all $a = s(\alpha)$, a is e-supported by $S \setminus \{x\}$, and (2) for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then either (2.1) there exists $\gamma \in \mathscr{R}$ where (2.1.1) $s(\gamma) \in S$ and either (2.1.2) $t(\gamma) = \beta$ or (2.1.3) $t(\gamma) = b$ for some $b = s(\beta)$, or (2.2) $\beta \notin \boxed{\mathscr{A} \cup \mathscr{R}}$ and there exists no $\delta \in \boxed{\varnothing}$ with (2.2.1) $t(\delta) = \beta$, (2.2.2) $s(\delta) \in \mathscr{A} \setminus Def(S)$, (2.2.3) δ is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, and (2.2.4) for all $c = s(\delta)$, c is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, or (2.3) for some $d = s(\beta)$, $d \notin \boxed{\mathscr{A} \cup \mathscr{R}}$ and there exists no $\delta \in \boxed{\varnothing}$ with (2.3.1) $t(\delta) = b$, (2.3.2) $s(\delta) \in \mathscr{A} \setminus Def(S)$, (2.3.3) δ is e-supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$, and (2.3.4) for all $c = s(\delta)$, c is supported by $\mathscr{A} \setminus (Def(S) \cup \{\beta\})$.

With these previous modifications, some conditions become obviously valid or contradictory. We rewrite the definition by replacing valid conditions with T and contradictory ones with F:

Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff (1.1) T or (1.2) F, and (2) for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then either (2.1) there exists $\gamma \in \mathscr{R}$ where (2.1.1) $s(\gamma) \in S$ and either (2.1.2) $t(\gamma) = \beta$ or (2.2.3) $t(\gamma) = b$ for some $b = s(\beta)$, or (2.2) F and T, or (2.3) F and T.

By deleting useless conditions, we obtain the following statement:

Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff for all $\beta \in \mathscr{R}$, if $t(\beta) = x$ then there exists $\gamma \in \mathscr{R}$ where $s(\gamma) \in S$ and either $t(\gamma) = \beta$ or $t(\gamma) = b$ for some $b = s(\beta)$.

Finally, applying (‖) gives (modifications are given in a box):

Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff for all $\boxed{y_\beta, z_\beta \text{ s.t. } y_\beta \mathscr{R} z_\beta}$, if $\boxed{z_\beta} = x$ then there exists $\boxed{y_\gamma, z_\gamma \text{ s.t. } y_\gamma \mathscr{R} z_\gamma}$ where $\boxed{y_\gamma} \in S$ and $\boxed{\sout{\text{either } z_\gamma = \beta \text{ or }}} \boxed{z_\gamma} = b$ for some $b = \boxed{y_\beta}$.

And so the final formulation (without any box) is:

Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and S a set of arguments. An element $x \in \mathscr{A}$ is acceptable wrt S iff for all y_β, z_β s.t. $y_\beta \mathscr{R} z_\beta$, if $z_\beta = x$ then there exists y_γ, z_γ s.t. $y_\gamma \mathscr{R} z_\gamma$ where $y_\gamma \in S$ and $z_\gamma = b$ for some $b = y_\beta$. □

In a nutshell, the semantics for Argumentation Frameworks (modulo the naming of attacks) are obtained from Definition 74 as follows.

Lemma 12. *Let $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF. A set of arguments S is said to be:*

- conflict-free *iff for all y, z s.t. $y \mathscr{R} z$, for all $x \in S$, if $y \in S$ then $z \neq x$,*
- admissible *iff S is conflict-free and for all $x \in S$, x is acceptable wrt U,*
- complete *iff S is admissible and for all $x \in \mathscr{A}$, if x is acceptable wrt S then $x \in S$,*
- preferred *iff S is a $\subseteq$-maximal admissible set of arguments,*
- grounded *iff S is a $\subseteq$-minimal complete set of arguments,*
- stable *iff $\mathscr{A} \setminus S = \{x \in \mathscr{A} \mid \exists y, z \text{ s.t. } y \mathscr{R} z,\ y \in S,\ z = x\}$.*

The equivalence of Lemma 12 with Definition 4 is straightforward.

4.4.4 Summary on Enriched Argumentation

Figure 4.1 displays the inclusion hierarchy that is induced by the enrichments considered in Section 2.3. That is to say, it displays the different frameworks that are obtained depending on which enrichments (or combinations thereof) are used.

Convention. As we discussed in the introduction, the term "Abstract Argumentation Framework" is then used to designate any type of framework in this hierarchy.

Table 4.2 provides a synopsis of the types of Abstract Argumentation Frameworks that are of interest in this work. The table includes for each type, its usual name, the name we use following our terminology, references, domain and their specificity (more precisely, the domain and co-domain of attacks and supports).

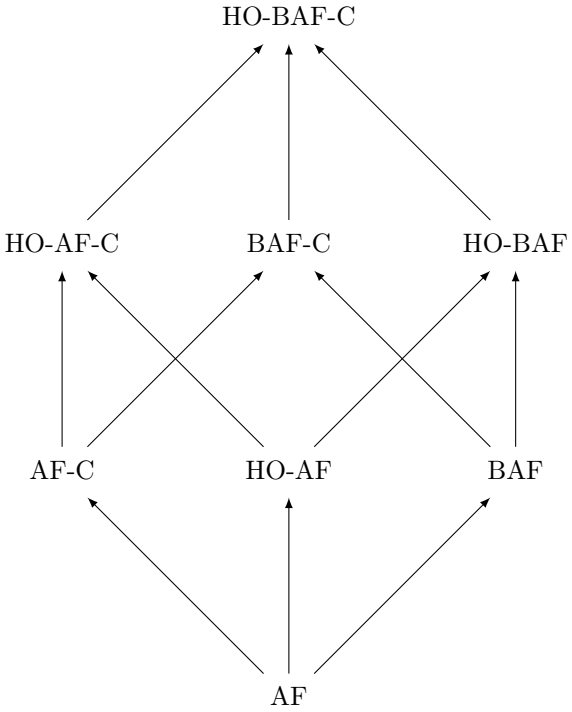

Figure 4.1: Hierarchy for classes of argumentation frameworks (inclusion of these classes)

4.5 A Family of Logical Theories for Enriched Abstract Argumentation

In this section, we present several logical theories that aim at capturing the different accounts of enriched Abstract Argumentation that are captured by the Higher-Order Evidence-Based Argumentation Frameworks with Coalitions formalism. More specifically, what we present is in fact a *generic logical theory* that can be adapted via a *parameterization* process to retrieve all the different frameworks captured by a Higher-Order Evidence-Based Argumentation Frameworks with Coalitions. The purpose of these logical theories is to give a way to compute some (or all) the extension(s) of some semantics in a given framework, typically using SAT solvers. We also hope that, with the way it is designed, it would not be very troublesome to extend the generic logical theory to capture more exotic semantics than the ones we consider in the present work.

Name		Usual name	References	Sets	Source function	Target function
AF		AF	[Dun95, BCG18]	$\mathscr{A}, \mathscr{R}$	$s : \mathscr{R} \rightarrow \mathscr{A}$	$t : \mathscr{R} \rightarrow \mathscr{A}$
AF-C		SETAF	[NP06, FB19]	$\mathscr{A}, \mathscr{R}$	$s : \mathscr{R} \rightarrow 2^{\mathscr{A}} \setminus \{\varnothing\}$	$t : \mathscr{R} \rightarrow \mathscr{A}$
HO-AF		AFRA/RAF	[BCGG11, CFFL21]	$\mathscr{A}, \mathscr{R}$	$s : \mathscr{R} \rightarrow \mathscr{A}$	$t : \mathscr{R} \rightarrow \mathscr{A} \cup \mathscr{R}$
HO-AF-C		N/A	[Lag23]	$\mathscr{A}, \mathscr{R}$	$s : \mathscr{R} \rightarrow 2^{\mathscr{A}} \setminus \{\varnothing\}$	$t : \mathscr{R} \rightarrow \mathscr{A} \cup \mathscr{R}$
BAF	AFD*	N/A	[BGvdTV10]	$\mathscr{A}, \mathscr{R}, \mathscr{S}[, \mathscr{P}]$	$s : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A}$	$t : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A}$
	AFN**	AFN	[NR11]			
	EBAF	EBAF	[ON08, OLR10]			
BAF-C	AFD-C	N/A	N/A	$\mathscr{A}, \mathscr{R}, \mathscr{S}[, \mathscr{P}]$	$s : \mathscr{R} \cup \mathscr{S} \rightarrow 2^{\mathscr{A}} \setminus \{\varnothing\}$	$t : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A}$
	AFN-C	AFN	[PO14]			
	EBAF-C	EBAF	[ON08, OLR10, PO14]			
HO-BAF	HO-AFD	N/A	[BGvdTV10]	$\mathscr{A}, \mathscr{R}, \mathscr{S}[, \mathscr{P}]$	$s : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A}$	$t : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$
	HO-AFN	ASAF	[CGGS15, GCGS18, Lag23]			
	HO-EBAF	N/A	N/A			
HO-BAF-C	HO-AFD-C	N/A	N/A	$\mathscr{A}, \mathscr{R}, \mathscr{S}[, \mathscr{P}]$	$s : \mathscr{R} \cup \mathscr{S} \rightarrow 2^{\mathscr{A}} \setminus \{\varnothing\}$	$t : \mathscr{R} \cup \mathscr{S} \rightarrow \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$
	HO-AFN-C	RAFN	[CFFL18b, Lag23]			
	HO-EBAF-C	REBAF	[CFFL18a, CFFL18b]			

Table 4.2: Range and combination of features for argumentation frameworks (N/A = Not Applicable).
Careful: the set $\mathscr{P}$ must be used when the meaning of the support is the evidential one
*: AFD refers to Abstract Argumentation Frameworks with a deductive interpretation of support
**: AFN refers to Abstract Argumentation Frameworks with a necessary interpretation of support

4.5.1 A Generic Theory

We begin by presenting the generic logical theory. Its axioms are formulas of first order logic with equality. Importantly, it is relative to a given Higher-Order Evidence-Based Argumentation Frameworks with Coalitions $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$, which is taken for granted throughout. Accordingly, we write $\mathfrak{L}_{Ext}(\mathcal{A})$ to denote the language of the theory.

Vocabulary

In our logical language, we will keep using the convention of using Latin letters to identify arguments and Greek letters to identify arcs (be they attacks or supports). As such, our individual variables will be from different fonts. Please keep in mind that this is for readability only: the logic is not two-sorted.

As to individual constants, instead of resorting to an explicit notation indicative of each element e in the given framework $\mathcal{A}$ being mapped to an individual constant specific to this element e, we adopt a more readable convention at the cost of abusing notation as follows.

Individual Constants For all e in $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$, we take e to be an individual constant.

As such, for a given element of the framework, the same letter will be used to designate it in the framework and to designate the logical constant that represents it in the logical language.

Predicates The list of unary and binary predicates, as well as their intended meaning, is presented in Table 4.3.

	Unary Predicates	**Meaning**
	$Arg(x)$	x is an argument
	$Att(x)$	x is an attack
	$Sup(x)$	x is a support
✓	$Cand(x)$	x can be a candidate for being selected by the semantics, *i.e.* for belonging to the result of this semantics (extension or structure)
	$PrimaFacie(x)$	x is a *prima facie* evidence
	$Selected(x)$	x is a member of the current extension/structure
✓	$Defeated(x)$	x is defeated by an attack
✓	$Inhibited(\alpha)$	α is inhibited by an attack
	$Acceptable(x)$	x is acceptable, in the sense of the defence wrt the attacks
	$Unacceptable(x)$	x cannot be acceptable
✓	$Activable(\alpha)$	the interaction α may be activated
✓	$Desactivated(\alpha)$	the interaction α cannot possibly be activated
	$Supported(x)$	x is supported
	$Unsupportable(x)$	x cannot be supported

	Binary Predicates	**Meaning**
	$S(\alpha, x)$	x is in the source of α
	$T(\alpha, x)$	x is in the target of α

Table 4.3: Unary and Binary Predicates (✓ indicates those which are what we call parameters of the theory)

We can make several observations on those predicates. First, please note that the last six unary predicates in this table can be viewed as having the same kind of behaviour:

- $Acceptable(x)$ is a necessary condition for x to be accepted although possibly not sufficient (depending on the characteristics of the argumentation framework and on the properties of the argumentation semantics under consideration). On the other hand, $Unacceptable(x)$ expresses a sufficient condition for x to fail to be accepted.

- *Activable*(α) is necessary for α to succeed (there is however no guarantee that α succeeds). In contrast, *Desactivated*(α) indicates that the attack α does fail.
- *Supported*(x) and $Unsupportable(x)$ work in a similar manner as *Acceptable*(x) and $Unacceptable(x)$ but concerning the notion of support. Thus, *Supported*(x) represents a necessary condition for x to receive evidential support while $Unsupportable(x)$ is intended to work as a sufficient condition for x not being able to receive such support.

The binary predicates S and T are obviously intended to capture the s and t functions of the argumentation framework $\mathcal{A}$. Although technically correct and consistent with the theory, the case where $t(\alpha)$ is a set of cardinality greater than 1 will not be considered in the rest of this work.

Importantly, we regard the unary predicates *Cand*, *Activable*, $Defeated$, *Inhibited* and *Desactivated* as **parameters of the generic theory**. They form the extra part supplementing the generic theory in order to capture the distinctive features of each enrichment, and thus of each type of framework.

Note. Notice that our language thus possesses a finite number of constant symbols, a finite number of predicate symbols and no function symbol.

Axioms for the Abstract Argumentation Framework

The formulas given here aim at describing the graph that constitutes the Abstract Argumentation Framework being handled, without the argumentative interpretation. That is to say, without encoding the effect of its relation(s) on the selection of its nodes

Since we are working with a given Higher-Order Evidence-Based Argumentation Frameworks with Coalitions $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$, it is convenient to fix the following notation for the elements of $\mathcal{A}$: We write $\mathscr{A} = \{e_1, \ldots, e_{n_1}\}$, $\mathscr{R} = \{e_{n_1+1}, \ldots, e_{n_2}\}$ and $\mathscr{S} = \{e_{n_2+1}, \ldots, e_{n_3}\}$ where $0 \leq n_1 \leq n_2 \leq n_3$. By convention, we assume that $n_1 = 0$ implies $\mathscr{A} = \varnothing$, $n_1 = n_2$ implies $\mathscr{R} = \varnothing$ and that $n_2 = n_3$ implies $\mathscr{S} = \varnothing$. In the case that $\mathcal{A}$ has a support relation, we write $\mathscr{P} = \{y_1, \ldots, y_k\}$, with the convention that $k = 0$ implies $\mathscr{P} = \varnothing$.

Axioms for the domain of elements of the framework

$$\text{for all } a \in \mathscr{A}, \quad Arg(a) \wedge \neg Att(a) \wedge \neg Sup(a) \tag{4.1a}$$
$$\text{for all } \alpha \in \mathscr{R}, \quad \neg Arg(\alpha) \wedge Att(\alpha) \wedge \neg Sup(\alpha) \tag{4.1b}$$
$$\text{for all } \alpha \in \mathscr{S}, \quad \neg Arg(\alpha) \wedge \neg Att(\alpha) \wedge Sup(\alpha) \tag{4.1c}$$
$$\text{for all } e_i, e_j \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S} \text{ s.t. } i \neq j, \quad \neg(e_i = e_j) \tag{4.1d}$$
$$\text{for all } e \in \mathscr{P}, \quad PrimaFacie(e) \tag{4.1e}$$
$$\text{for all } e \in (\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}) \setminus \mathscr{P}, \quad \neg PrimaFacie(e) \tag{4.1f}$$

Closure axioms Considering $\mathscr{A} = \{e_1, \ldots, e_{n_1}\}$, $\mathscr{R} = \{e_{n_1+1}, \ldots, e_{n_2}\}$ and $\mathscr{S} = \{e_{n_2+1}, \ldots, e_{n_3}\}$,

$$\forall z \ \big(Arg(z) \vee Att(z) \vee Sup(z)\big) \tag{4.2a}$$
$$\forall z \ \big(Arg(z) \to z = e_1 \vee \cdots \vee z = e_{n_1}\big) \tag{4.2b}$$
$$\forall z \ \big(Att(z) \to z = e_{n_1+1} \vee \cdots \vee z = e_{n_2}\big) \tag{4.2c}$$
$$\forall z \ \big(Sup(z) \to z = e_{n_2+1} \vee \cdots \vee z = e_{n_3}\big) \tag{4.2d}$$

Remark. An immediate consequence of (4.2) is

$$\forall z \ \big(z = e_1 \vee \cdots \vee z = e_{n_3}\big) \tag{UDC$_\mathcal{A}$}$$

where UDC stands for: *Usual Domain Closure*. In particular, we can restrict ourselves to interpretations in which the domain is precisely $\mathscr{A} \cup \mathscr{R} \cup \S$, and so is finite.

Axiom for attacks and supports

$$\text{for all } \alpha \in \mathscr{R} \cup \mathscr{S}, \quad \Big(\bigwedge_{a \in s(\alpha)} S(\alpha, a) \Big) \wedge \Big(\bigwedge_{e \in t(\alpha)} T(\alpha, e) \Big) \tag{4.3}$$

Remark. Please observe that the theory is non-contradictory ($\mathscr{A}$, $\mathscr{R}$, and $\mathscr{S}$ are pairwise disjoint).

Note. Please note that each of Axioms (4.1) produces several formulas (one for each concerned element). The same thing happens for Axiom (4.3) whereas it is not the case for Axioms (4.2) or (UDC$_{\mathcal{A}}$). Indeed, in this last case, each axiom produces only one formula containing a quantifier over z that is a variable of the language. This difference is essential and is formalised in the text, either with "for all" (case of Axioms (4.1) and (4.3)), or "$\forall$" (case of Axioms (4.2) or (UDC$_{\mathcal{A}}$)). In the rest of the document, the use of "$\forall$" means that the corresponding formula contains this quantifier and, in this case, the domain of the used variable is given by the domain associated to the language (that is defined of course using $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$).

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\mathcal{S}_{\mathcal{A}}$ the subtheory consisting of Axioms (4.1) and (4.3) together, and by $\Sigma(\mathcal{A})$ the subtheory consisting of Axioms (4.1), (4.2) and (4.3) together.

In the following, statements refer to Herbrand models whence the next theorem that provides a correspondence between Herbrand models and arbitrary models for the theory. Recall that, given a formula ϕ, a Herbrand interpretation for ϕ is any set of ground atoms relevant for ϕ, and given a set of formulas Σ, a Herbrand model for Σ is a Herbrand interpretation that satisfies all formulas in Σ.

Theorem 11. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. Let φ be a formula of $\mathfrak{L}_{\mathcal{A}}$. $\mathcal{S}_{\mathcal{A}}, UDC_{\mathcal{A}} \models \varphi$ if and only if $\varphi \in \mathrm{Th}(\mathfrak{H}_{\mathcal{S}_{\mathcal{A}}})$.*

Basic Principles Underlying Argumentation Semantics

The formulas given here aim at describing the argumentative interpretation of the graph. That is to say, these formulas encode semantics to select arguments. More precisely, we rely on the decomposition of semantics into underlying principles, as presented in Section 2.4. Thus, our formulas encode in fact those underlying principles.

We begin with the formula that encodes which elements are selected as a structure (or extension), the formulas that encode the Coherence principle, and those that encode the interpretation of the support relation.

Acceptability

$$\forall x \ \big(\mathit{Selected}(x) \leftrightarrow (\mathit{Acceptable}(x) \wedge \mathit{Supported}(x)) \big) \tag{4.4}$$

Coherence

$$\forall \alpha \in \mathit{Att} \ \Big(\mathit{Activable}(\alpha) \rightarrow \exists x \in \mathit{Cand} \ \big(T(\alpha, x) \wedge \mathit{Unacceptable}(x) \big) \Big) \tag{4.5a}$$

$$\forall x \in \mathit{Cand} \ \big(\mathit{Unacceptable}(x) \rightarrow \neg \mathit{Acceptable}(x) \big) \tag{4.5b}$$

Remark. Strictly speaking and in line with our previous discussions, formulas (4.5) do not rigorously capture the Coherence principle, because predicates *Activable* and *Cand* have not yet been axiomatized (it will be done below for each class of argumentation frameworks).

Note. In Formula (4.5a), the conclusion of the implication could also be encoded by $\forall x \in \mathit{Cand}\ (T(\alpha, x) \rightarrow \mathit{Unacceptable}(x))$. Since in the present work we restrict ourselves to the cases where interactions can only have one target, it does not change anything for us. Without this restriction, it would lead to a different interpretation of sets of elements as targets.

Support

$$\forall x \in Cand$$
$$\Big(\Big(PrimaFacie(x) \vee \exists \alpha \in Sup \ \big(T(\alpha, x) \wedge Activable(\alpha)\big)\Big) \rightarrow Supported(x)\Big) \tag{4.6}$$

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\Sigma_{Coh}(\mathcal{A})$ the subtheory consisting of $\Sigma(\mathcal{A})$ and Axioms (4.4), (4.5) and (4.6) together.

$\Sigma_{Coh}(\mathcal{A})$ can be seen as a sort of core theory. It encapsulates both the encoding of the framework at hand, and the first bricks (i.e. underlying principles) used to define argumentative semantics. In accordance with how semantics are decomposed (cf. Section 2.4), it should be supplemented with additional axioms, representing additional principles, to capture more complex semantics.

The next formulas encode the Self-support principle, necessary to properly decompose semantics in the case where an evidential support relation is present.

Self-support

$$\forall x \in Cand \ \Big(Supported(x) \rightarrow \Big(PrimaFacie(x) \vee \exists \alpha \in Sup \ \big(T(\alpha, x) \wedge Activable(\alpha)\big)\Big)\Big) \tag{4.7a}$$

$$\forall x \in Cand \ \Big(Unsupportable(x) \leftrightarrow \Big(\neg PrimaFacie(x) \wedge \forall \alpha \in Sup \ \big(T(\alpha, x) \rightarrow Desactivated(\alpha)\big)\Big)\Big) \tag{4.7b}$$

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\Sigma_{SS}(\mathcal{A})$ the subtheory consisting of $\Sigma_{Coh}(\mathcal{A})$ and Axioms (4.7) together.

Then, comes the formula that encodes the Defence principle.

Defence

$$\forall x \in Cand \ \Big(Acceptable(x) \rightarrow \Big(\forall \alpha \in Att \ \big(T(\alpha, x) \rightarrow Desactivated(\alpha)\big)\Big)\Big) \tag{4.8}$$

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\Sigma_{Def}(\mathcal{A})$ the subtheory consisting of $\Sigma_{SS}(\mathcal{A})$ and Axiom (4.8) together.

The next formula encodes the Reinstatement principle.

Reinstatement

$$\forall x \in Cand \ \Big(\Big(\forall \alpha \in Att \ \big(T(\alpha, x) \rightarrow Desactivated(\alpha)\big)\Big) \rightarrow Acceptable(x)\Big) \tag{4.9}$$

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\Sigma_{Rein}(\mathcal{A})$ the subtheory consisting of $\Sigma_{Def}(\mathcal{A})$ and Axiom (4.9) together.

Lastly, a couple of axioms is dedicated to the Complement Attack principle.

Complement attack

$$\forall x \in Cand \ \big(\neg Acceptable(x) \rightarrow Defeated(x)\big) \tag{4.10a}$$

$$\forall x \in Cand \ \big(\neg Supported(x) \rightarrow Unsupportable(x)\big) \tag{4.10b}$$

Notation. In the following, considering an HO-EBAF-C $\mathcal{A}$, we denote by $\Sigma_{CA}(\mathcal{A})$ the subtheory consisting of $\Sigma_{SS}(\mathcal{A})$ and Axioms (4.10) together.

Semantics

We now give the links between the structures (or extensions) of an Abstract Argumentation Framework and the models of its logical encoding. The following definitions allow us to retrieve a structure from a model, and to have a notion corresponding to the Maximality and Minimality principles in the logical encoding.[3]

Definition 75. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. Let I be an interpretation over $\mathfrak{L}_{Ext}(\mathcal{A})$. We define

- $S_I = \{a \in \mathscr{A} \mid I(Selected(a)) = \top\}$
- $\Gamma_I = \{\alpha \in \mathscr{R} \mid I(Selected(\alpha)) = \top\}$
- $\Delta_I = \{\alpha \in \mathscr{S} \mid I(Selected(\alpha)) = \top\}$

Definition 76. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. Let I be a model of a set of formulas Π over $\mathfrak{L}_{Ext}(\mathcal{A})$.

- I is a $\subseteq$-*maximal* model of Π if and only if there is no model I' of Π such that $(S_I \cup \Gamma_I \cup \Delta_I) \subset (S_{I'} \cup \Gamma_{I'} \cup \Delta_{I'})$
- I is a $\subseteq$-*minimal* model of Π if and only if there is no model I' of Π such that $(S_{I'} \cup \Gamma_{I'} \cup \Delta_{I'}) \subset (S_I \cup \Gamma_I \cup \Delta_I)$

Consider a structure $U = (S, \Gamma, \Delta)$ for $\mathcal{A}$ (Definition 68). We are to characterize σ-structures for $\mathcal{A}$ by verifying the following properties for each class of Abstract Argumentation Frameworks. This approach is very close to the work done in [CL18, CL20].

Remark. The properties that are to come are expressed relative to a varying set of formulas Σ', which is meant to encode features distinctive of each particular type of enrichment. Hence, the following property is a generic property parameterized by Σ': it must be instantiated in order to be applied to each studied framework.

Property 1. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and $U = (S, \Gamma, \Delta)$ be a structure for $\mathcal{A}$. Let Σ' be a set of formulas over $\mathfrak{L}_{Ext}(\mathcal{A})$.*

1. *U is* conflict-free *if and only if there exists a model I of $\Sigma_{Coh}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
2. *U is* admissible *if and only if there exists a model I of $\Sigma_{Def}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
3. *U is* complete *if and only if there exists a model I of $\Sigma_{Rein}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
4. *U is* preferred *if and only if there exists a model I $\subseteq$-maximal of $\Sigma_{Def}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
5. *U is* grounded *if and only if there exists a model I $\subseteq$-minimal of $\Sigma_{Rein}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
6. *U is* stable *if and only if there exists a model I of $\Sigma_{CA}(\mathcal{A}) \cup \Sigma'$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*

Note. The case of structures is only correct for Abstract Argumentation Frameworks with higher-order relations with the RAF interpretation. In other cases, we consider, instead of such a structure, a single extension containing all accepted elements, these being arguments or even attacks or supports in some cases. The idea to characterize a σ-extension T of an Abstract Argumentation Framework is the same as with a σ-structure $U = (S, \Gamma, \Delta)$, except that in the case of T, the condition $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$ is replaced by $T = S_I \cup \Gamma_I \cup \Delta_I$.

[3] Observe that we gave no axiom to encode the Maximality and Minimality principles.

The point is now to specify Σ' for each type of argumentation framework so that Properties 1.1 to 1.6 hold for the type under consideration.

4.5.2 Simplification and Specialisations

Before instantiating the parameters of the theory for every kind of enrichment (and their combinations) so that Properties 1.1 to 1.6 hold for them, we present in this section some methods whose aim is to simplify or constrain the generic logical theory in order to adapt it to specific cases of Abstract Argumentation Frameworks.

Simplification: No Support

The first method simplifies the theory for Abstract Argumentation Frameworks without a support relation. In practice in this case, various formulas reduce to much simpler ones, or are even trivially satisfied.

The first result states that, under such circumstances, Formula (4.6) is in fact equivalent to all elements being considered supported.

Proposition 6. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ that satisfies $\forall x \ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.6) if and only if it is a model of $\forall x \in Cand \ (Supported(x))$.*

In the case of Formula (4.7a), it is trivially satisfied provided that Formula (4.6) is as well.

Proposition 7. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ and (4.6) that satisfies $\forall x \ (Cand(x) \rightarrow (Arg(x) \vee Att(x)))$ is a model of (4.7a), i.e., $\forall x \in Cand \ \big(Supported(x) \rightarrow [PrimaFacie(x) \vee \exists \alpha \in Sup \ (T(\alpha, x) \wedge Activable(\alpha))]\big)$*

Similarly to the first result, Formula (4.7b) is in fact equivalent to all elements being considered not unsupportable.

Proposition 8. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, each Herbrand model of $\Sigma(\mathcal{A})$ and (4.6) that satisfies $\forall x \ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.7b) if and only if it is a model of $\forall x \in Cand \ (\neg Unsupportable(x))$.*

Lastly, Formula (4.10b) is trivially satisfied as well.

Proposition 9. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ that satisfies $\forall x \ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.10b), i.e., $\forall x \in Cand \ \big(\neg Supported(x) \rightarrow Unsupportable(x)\big)$*

Remark. The alert reader may have noticed that all these propositions are relative to some instantiating of the parameter $Cand$. As we will see, every type of Abstract Argumentation Framework has in fact this parameter instantiated by a formula that captures the one used as instantiating in these propositions.

Specialisation 1: Sources Are Singletons Sets

In order to impose the constraint that each source (of an attack or a support) is a single argument, the theory at hand must be supplemented with the following axiom:

$$\forall \alpha \in Att \cup Sup \ \Big(\forall a \in Arg \ \big(\forall b \in Arg \ [(S(\alpha, a) \wedge S(\alpha, b)) \rightarrow a = b]\big)\Big) \tag{4.11}$$

Specialisation 2: Targets Are Singletons Sets

In order to impose the constraint that each target (of an attack or a support) is a single argument, the theory at hand must be supplemented with the following axiom:

$$\forall \alpha \in Att \cup Sup \ \Big(\forall a \in Arg \ \big(\forall b \in Arg \ [(T(\alpha, a) \wedge T(\alpha, b)) \rightarrow a = b] \big) \Big) \tag{4.12}$$

As we only deal with the case of a single target throughout the present work, please keep in mind that Axiom (4.12) is present in the theory whatever case we consider.

4.5.3 Theory for an Argumentation Framework

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to Argumentation Frameworks (see Section 2.1). For an Argumentation Framework, the parameters of the generic theory are axiomatized as follows.

Parameters for an AF

$$\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \big) \tag{4.13a}$$
$$\forall \alpha \ \big(Activable(\alpha) \leftrightarrow \big(\forall a \in Arg \ [S(\alpha, a) \rightarrow Selected(a)] \big) \big) \tag{4.13b}$$
$$\forall x \ \big(Defeated(x) \leftrightarrow \big(\exists \alpha \in Att \ [T(\alpha, x) \wedge Activable(\alpha)] \big) \big) \tag{4.13c}$$
$$\forall \alpha \ \big(Inhibited(\alpha) \leftrightarrow \big(\exists a \in Arg \ [S(\alpha, a) \wedge Defeated(a)] \big) \big) \tag{4.13d}$$
$$\forall \alpha \ \big(Desactivated(\alpha) \leftrightarrow Inhibited(\alpha) \big) \tag{4.13e}$$

Formula (4.13a) specifies that the only elements of the Argumentation Framework that can be part of an extension are the arguments. Formula (4.13b) tells us that an attack is activable if and only if all the arguments of its source are accepted in the extension. Then, Formula (4.13c) describes a defeated element as an element that is targeted by an activable attack. Using this, Formula (4.13d) identifies inhibition with the presence of a defeated element in the source of the inhibited element. Finally, Formula (4.13e) says that being inhibited coincides with desactivation.

Note that (4.13) can be used to rewrite formulas (4.5), (4.8), (4.9) and (4.10a) as formulas $(4.5a)_{\text{AF}}$, $(4.5b)_{\text{AF}}$, $(4.8)_{\text{AF}}$, $(4.9)_{\text{AF}}$, $(4.10)_{\text{AF}}$ below.

Conflict-free

$$\forall \alpha \in Att \ \Big(\big(\forall a \in Arg \ [S(\alpha, a) \rightarrow Selected(a)] \big) \rightarrow \big(\exists x \in Arg \ [T(\alpha, x) \wedge Unacceptable(x)] \big) \Big) \tag{$(4.5a)_{\text{AF}}$}$$
$$\forall x \in Arg \ \big(Unacceptable(x) \rightarrow \neg Acceptable(x) \big) \tag{$(4.5b)_{\text{AF}}$}$$

Defence

$$\forall x \in Arg \ \Big(Acceptable(x) \rightarrow \ \forall \alpha \in Att \ \Big(T(\alpha, x) \rightarrow \ \exists a \in Arg \ \Big[S(a, \alpha) \wedge \exists \beta \in Att \ \Big(T(\beta, a) \wedge \forall b \in Arg \ [S(\beta, b) \rightarrow Selected(b)] \Big) \Big] \Big) \Big) \tag{$(4.8)_{\text{AF}}$}$$

Reinstatement

$$\forall x \in Arg \ \Big(\Big[\forall \alpha \in Att \ \Big(T(\alpha, x) \rightarrow \exists a \in Arg \ \Big(S(a, \alpha) \wedge \exists \beta \in Att \ \big(T(\beta, a) \wedge \forall b \in Arg \ [S(\beta, b) \rightarrow Selected(b)]\big)\Big)\Big)\Big] \rightarrow Acceptable(x)\Big) \quad ((4.9)_{\text{AF}})$$

Complement attack

$$\forall x \in Arg \ \Big(\neg Acceptable(x) \rightarrow \exists \alpha \in Att \ \Big(T(\alpha, x) \wedge \big(\forall a \in Arg \ [S(\alpha, a) \rightarrow Selected(a)]\big)\Big)\Big) \quad ((4.10\text{a})_{\text{AF}})$$

Note. Note that Formula $(4.5\text{a}_{\text{AF}})$ can also be rewritten as:

$$\forall x \in Cand \ \Big(\ Defeated(x) \rightarrow Unacceptable(x)\ \Big) \quad ((4.5\text{a})_{\text{AF}})$$

Properties 1.1 to 1.6 hold whenever $\mathcal{A}$ is an Argumentation Framework using formulas (4.13), as well as Axioms 4.11 and (4.12) as Σ'. This is captured by the following proposition that matches the results in [CL18].

Proposition 10. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an AF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

Remark. Observe that since $S \subseteq \mathscr{A}$, we necessarily have $\Gamma_I = \Delta_I = \varnothing$ in the previous proposition.

4.5.4 Theory for an Argumentation Framework with Coalitions (AF-C)

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to Argumentation Frameworks with Coalitions (see Section 2.3.1). For an Argumentation Framework with Coalitions, the parameters of the generic theory are axiomatized as follows.

Parameters for an AF-C

$$\forall x \ \big(Cand(x) \leftrightarrow Arg(x)\big) \tag{4.14a}$$

$$\forall \alpha \ \Big(Activable(\alpha) \leftrightarrow \Big(\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big)\Big)\Big) \tag{4.14b}$$

$$\forall x \ \Big(Defeated(x) \leftrightarrow \Big(\exists \alpha \in Att\big(\ T(\alpha, x) \wedge Activable(\alpha)\ \big)\Big)\Big) \tag{4.14c}$$

$$\forall \alpha \ \Big(Inhibited(\alpha) \leftrightarrow \Big(\exists a \in Arg\big(\ S(\alpha, a) \wedge Defeated(a)\ \big)\Big)\Big) \tag{4.14d}$$

$$\forall \alpha \ \big(Desactivated(\alpha) \leftrightarrow Inhibited(\alpha)\big) \tag{4.14e}$$

Remark. The parameters are axiomatized in the same way as for Argumentation Frameworks (see Section 4.5.3), and thus have the same meaning. The only difference between the two theories is the presence, in the case of an Argumentation Framework, of the axiom (4.11) which imposes that each source is a single argument. This implies two observations. Firstly, the fundamental behaviour of Argumentation Frameworks, with or without Coalitions, is the same. Secondly, Argumentation Frameworks are a special case of Argumentation Frameworks with Coalitions.

Note. Since formulas (4.14) are the same as formulas (4.13), we will refer to the latter.

Since the parameters of an Argumentation Framework with Coalitions are the same as an Argumentation Framework, formulas (4.5), (4.8), (4.9) and (4.10a) can be rewritten as in Section 4.5.3. We will thus refer to formulas $(4.5a)_{\text{AF}}$, $(4.5b)_{\text{AF}}$, $(4.8)_{\text{AF}}$, $(4.9)_{\text{AF}}$, $(4.10)_{\text{AF}}$, even in the case of Argumentation Framework with Coalitions.

Properties 1.1 to 1.6 hold whenever $\mathcal{A}$ is an Argumentation Framework with Coalitions using formulas (4.13), as well as Axiom (4.12) as Σ'. This is captured by the following proposition.

Proposition 11. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an AF-C and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

Remark. This behavior of simply removing Axiom (4.11) from the theory of an Abstract Argumentation Framework and obtaining a theory that captures the same framework with Coalitions is in fact transferable to all type of Abstract Argumentation Framework considered in the present work. Thus, in the following, we will only present frameworks in which Coalitions are absent, the theory capturing the case where they are present being effectively the same in which we remove Axiom (4.11).

4.5.5 Theory for a Higher-Order Argumentation Framework (HO-AF)

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to Higher-Order Argumentation Frameworks (see Section 2.3.2). For a Higher-Order Argumentation Framework, the parameters of the generic theory are axiomatized as follows.

Parameters for an HO-AF

$$\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \vee Att(x)\big) \tag{4.15a}$$

$$\forall \alpha \ \Big(Activable(\alpha) \leftrightarrow \Big(\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge Selected(\alpha)\Big)\Big) \tag{4.15b}$$

$$\forall x \ \Big(Defeated(x) \leftrightarrow \Big(\exists \alpha \in Att\big(\ T(\alpha, x) \wedge Activable(\alpha)\ \big)\Big)\Big) \tag{4.15c}$$

$$\begin{aligned}\forall \alpha \ \Big(&Inhibited(\alpha) \leftrightarrow \\ &\Big(\exists \beta \in Att\Big(\ \Big(\exists a \in Arg\big(\ S(\alpha, a) \wedge T(\beta, a)\ \big) \vee T(\beta, \alpha)\Big) \wedge Activable(\beta)\ \Big)\Big)\Big)\end{aligned} \tag{4.15d}$$

$$\forall \alpha \ \big(Desactivated(\alpha) \leftrightarrow Inhibited(\alpha)\big) \tag{4.15e}$$

Formulas (4.15) differ from formulas (4.13) in several points. To begin with, Formula (4.15a) specifies that both arguments and attacks can be part of a structure. Then, Formula (4.15b) tells us that to be activable, an interaction must have all the arguments of its source accepted, as in Formula (4.13b), but in addition the interaction itself must also be accepted. Using this, Formula (4.15d) says that an element is inhibited if and only if there is an attack that targets either the element or one argument of its source and that is activable. Please note that this is equivalent to

$$\forall \alpha(\ Inhibited(\alpha) \leftrightarrow (Defeated(\alpha) \vee \exists a \in Arg(\ S(\alpha, a) \wedge Defeated(a)\ )\ ))$$

As expected, we now present formulas (4.5), (4.8), (4.9) and (4.10) rewritten using formulas (4.15).

Conflict-free

$$\begin{aligned}\forall \alpha \in Att \ \Big(\ \Big[\forall a \in Arg \ \big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge Selected(\alpha)\Big] \rightarrow \\ \exists x \in (Arg \cup Att) \ \big(T(\alpha, x) \wedge Unacceptable(x)\big)\ \Big)\end{aligned} \tag{$(4.5a)_{\text{HOAF}}$}$$

$$\forall x \in (Arg \cup Att) \ \big(\ Unacceptable(x) \rightarrow \neg Acceptable(x)\ \big) \tag{$(4.5b)_{\text{HOAF}}$}$$

Defence

$$\begin{aligned}\forall x \in (Arg \cup Att) \ \Big(\ &Acceptable(x) \rightarrow \forall \alpha \in Att \ \Big(\ T(\alpha, x) \rightarrow \\ &\exists \beta \in Att \ \Big(\ \big[\exists a \in Arg \ \big(\ S(\alpha, a) \wedge T(\beta, a)\ \big) \vee T(\beta, \alpha)\big] \wedge \\ &\big[\forall b \in Arg \ \big(\ S(\beta, b) \rightarrow Selected(b)\ \big) \wedge Selected(\beta)\big]\ \Big)\ \Big)\ \Big)\end{aligned} \tag{$(4.8)_{\text{HOAF}}$}$$

Reinstatement

$$\forall x \in (Arg \cup Att) \ \Bigg(\forall \alpha \in Att \Big(T(\alpha, x) \rightarrow \exists \beta \in Att \ \Big(\big[\exists a \in Arg \ \big(S(\alpha, a) \wedge T(\beta, a) \big) \vee T(\beta, \alpha) \big] \wedge \big[\forall b \in Arg \ \big(S(\beta, b) \rightarrow Selected(b) \big) \wedge Selected(\beta) \big] \Big) \Big) \rightarrow Acceptable(x) \Bigg) \quad ((4.9)_{\text{HOAF}})$$

Complement attack

$$\forall x \in (Arg \cup Att) \ \Big(\neg Acceptable(x) \rightarrow \exists \alpha \in Att \ \Big(T(\alpha, x) \wedge \forall a \in Arg \ \big(S(\alpha, a) \rightarrow Selected(a) \big) \wedge Selected(\alpha) \Big) \Big) \quad ((4.10a)_{\text{HOAF}})$$

Note. Note that Formula $(4.5a_{\text{HOAF}})$ can be rewritten in the same way as Formula $(4.5a_{\text{AF}})$ in Section 4.5.3.

Properties 1.1 to 1.6 hold whenever $\mathcal{A}$ is a Higher-Order Argumentation Framework using formulas (4.15), as well as Axioms 4.11 and (4.12) as Σ'. This is captured by the following proposition.

Proposition 12. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an HO-AF and $U = (S, \Gamma, \Delta)$ be a structure.*

1. *U is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
2. *U is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
3. *U is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
4. *U is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
5. *U is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
6. *U is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*

4.5.6 Theory for an Evidence-Based Argumentation Framework (EBAF)

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to Evidence-Based Argumentation Frameworks (see Section 2.3.3). For an Evidence-Based Argumentation Framework, the parameters of the generic theory are axiomatized as follows.

Please remember that, in this work, we only take into account Evidence-Based Argumentation Frameworks without support cycles. That is due to the constraint in e-support that an element cannot support

itself. Note also that this restriction has been lifted in [Lag23], not by changing the logical formulas, but by adding a constraint on the logical models we consider.

Remember as well that according to our naming conventions, we do not consider Evidence-Based Argumentation Frameworks to have relations between sets of arguments (that would be Evidence-Based Argumentation Frameworks with Coalitions), while EBAFs in the literature do have this functionality.

Parameters for an EBAF

$$\forall x \ \big(Cand(x) \leftrightarrow Arg(x)\big) \tag{4.16a}$$

$$\forall \alpha \ \Big(Activable(\alpha) \leftrightarrow \Big(\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big)\Big)\Big) \tag{4.16b}$$

$$\forall x \ \Big(Defeated(x) \leftrightarrow \Big(\exists \alpha \in Att\big(\ T(\alpha, x) \wedge Activable(\alpha)\ \big)\Big)\Big) \tag{4.16c}$$

$$\forall \alpha \ \Big(Inhibited(\alpha) \leftrightarrow \Big(\exists a \in Arg\big(\ S(\alpha, a) \wedge Defeated(a)\ \big)\Big)\Big) \tag{4.16d}$$

$$\begin{aligned} \forall \alpha \ \Big(&Desactivated(\alpha) \leftrightarrow \\ &\Big(\exists a \in Arg\big(\ S(\alpha, a) \wedge Unsupportable(a)\ \big) \vee Inhibited(\alpha)\Big)\Big) \end{aligned} \tag{4.16e}$$

Formulas (4.16) are very similar to formulas (4.13). The only difference is that in the case of Evidence-Based Argumentation Frameworks, the deactivation of an interaction is not strictly the same as its inhibition. It is also sufficient to make one argument of its source unsupportable.

Since a support relation is present in Evidence-Based Argumentation Frameworks, we cannot use propositions 6 to 9. Thus, we present formulas (4.5), (4.6), (4.7), (4.8), (4.9) and (4.10) rewritten using formulas (4.16).

Conflict-free

$$\begin{aligned} \forall \alpha \in Att\Big(\ &\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \rightarrow \\ &\exists x \in Arg\big(T(\alpha, x) \wedge Unacceptable(x)\big)\ \Big) \end{aligned} \tag{$(4.5a)_{\text{EBAF}}$}$$

$$\forall x \in Arg\big(\ Unacceptable(x) \rightarrow \neg Acceptable(x)\ \big) \tag{$(4.5b)_{\text{EBAF}}$}$$

Support

$$\begin{aligned} \forall x \in Cand\Big(\ &\Big[PrimaFacie(x) \vee \\ &\exists \alpha \in Sup\Big(\ T(\alpha, x) \wedge \forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big)\ \Big)\Big] \\ &\rightarrow Supported(x)\ \Big) \end{aligned} \tag{$(4.6)_{\text{EBAF}}$}$$

Self-support

$$\forall x \in Cand\Big(\; Supported(x) \rightarrow \Big[PrimaFacie(x) \vee \exists \alpha \in Sup\Big(\; T(\alpha, x) \wedge \forall a \in Arg\big(\; S(\alpha, a) \rightarrow Selected(a) \; \big) \Big) \Big] \; \Big) \tag{(4.7a)$_{\text{EBAF}}$}$$

$$\forall x \in Cand\Big(\; Unsupportable(x) \leftrightarrow \Big[\neg PrimaFacie(x) \wedge \forall \alpha \in Sup\Big(\; T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(\; S(\alpha, a) \wedge Unsupportable(a) \; \big) \vee \exists a \in Arg\big(\; S(\alpha, a) \wedge \exists \beta \in Att\big(\; T(\beta, a) \wedge \forall b \in Arg(\; S(\beta, b) \rightarrow Selected(b) \;) \; \big) \big) \Big] \Big) \Big] \; \Big) \tag{(4.7b)$_{\text{EBAF}}$}$$

Defence

$$\forall x \in Cand\Big(\; Acceptable(x) \rightarrow \forall \alpha \in Att\Big(\; T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(\; S(\alpha, a) \wedge Unsupportable(a) \; \big) \vee \exists a \in Arg\big(\; S(\alpha, a) \wedge \exists \beta \in Att\big(\; T(\beta, a) \wedge \forall b \in Arg(\; S(\beta, b) \rightarrow Selected(b) \;) \; \big) \big) \Big] \Big) \Big) \tag{(4.8)$_{\text{EBAF}}$}$$

Reinstatement

$$\forall x \in Cand\Big(\; \forall \alpha \in Att\Big(\; T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(\; S(\alpha, a) \wedge Unsupportable(a) \; \big) \vee \exists a \in Arg\big(\; S(\alpha, a) \wedge \exists \beta \in Att\big(\; T(\beta, a) \wedge \forall b \in Arg(\; S(\beta, b) \rightarrow Selected(b) \;) \; \big) \big) \Big] \Big) \rightarrow Acceptable(x) \Big) \tag{(4.9)$_{\text{EBAF}}$}$$

Complement attack

$$\forall x \in Cand\Big(\; \neg Acceptable(x) \rightarrow \exists \alpha \in Att\big(\; T(\alpha, x) \wedge \forall a \in Arg\big(\; S(\alpha, a) \rightarrow Selected(a) \; \big) \; \Big) \tag{(4.10a)$_{\text{EBAF}}$}$$

$$\forall x \in Cand\big(\; \neg Supported(x) \rightarrow Unsupportable(x) \; \big) \tag{(4.10b)$_{\text{EBAF}}$}$$

Note. Again, Formula ($4.5a_{\text{EBAF}}$) can be rewritten in the same way as Formula ($4.5a_{\text{AF}}$) in Section 4.5.3.

Properties 1.1 to 1.6 hold whenever $\mathcal{A}$ is an Evidence-Based Argumentation Framework using formulas (4.16), as well as Axioms 4.11 and (4.12) as Σ'. This is captured by the following proposition.

Proposition 13. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an EBAF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

4.5.7 Theory for a Higher-Order Evidence-based Argumentation Framework (HO-EBAF)

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to Higher-Order Evidence-Based Argumentation Frameworks. For a Higher-Order Evidence-Based Argumentation Framework, the parameters of the generic theory are axiomatized as follows.

As mentioned before, we only take into account Higher-Order Evidence-based Argumentation Framework without support cycles.

Parameters for a Higher-Order Evidence-Based Argumentation Framework

$$\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \vee Att(x) \vee Sup(x)\big) \tag{4.17a}$$

$$\forall \alpha \ \Big(Activable(\alpha) \leftrightarrow \Big(\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge Selected(\alpha)\Big)\Big) \tag{4.17b}$$

$$\forall x \ \Big(Defeated(x) \leftrightarrow \Big(\exists \alpha \in Att\big(\ T(\alpha, x) \wedge Activable(\alpha)\ \big)\Big)\Big) \tag{4.17c}$$

$$\begin{aligned} &\forall \alpha \ \Big(Inhibited(\alpha) \leftrightarrow \\ &\exists \beta \in Att\Big(\big[T(\beta, \alpha) \vee \exists a \in Arg(S(\alpha, a) \wedge T(\beta, a))\big] \wedge Activable(\beta)\Big)\Big) \end{aligned} \tag{4.17d}$$

$$\begin{aligned} &\forall \alpha \ \Big(Desactivated(\alpha) \leftrightarrow \\ &\quad \Big(\exists a \in Arg\big(\ S(\alpha, a) \wedge Unsupportable(a)\ \big) \\ &\quad \vee Unsupportable(\alpha) \vee Inhibited(\alpha)\Big)\Big) \end{aligned} \tag{4.17e}$$

Please observe that formulas (4.17) are in fact the combination of formulas (4.15) and (4.16). Thus, Formula (4.17a) specifies that arguments, attacks and supports can be part of a structure. Then, Formula (4.17b) tells us that to be activable, an interaction must be accepted and have all the arguments of its source accepted, as in Formula (4.15b). Using this, Formula (4.17d) says that an element is inhibited if and only if there is an attack that targets either the element or one argument of its source and that is activable, as in Formula (4.15d). Finally, Formula (4.17e) states that to be deactivated, an interaction must be inhibited, or unsupportable, or have an an argument of its source being unsupportable. This in fact integrating the higher-order relations generalisation into Formula (4.16e).

Again, since a support relation is present in Higher-Order Evidence-Based Argumentation Frameworks, we cannot use propositions 6 to 9. Thus, we present formulas (4.5), (4.6), (4.7), (4.8), (4.9) and (4.10) rewritten using formulas (4.17).

Conflict-free

$$\forall \alpha \in Att\Big(\Big[\forall a \in Arg\big(S(\alpha, a) \rightarrow Selected(a) \big) \wedge Selected(\alpha)\Big] \rightarrow \exists x \in (Arg \cup Att \cup Sup)\big(T(\alpha, x) \wedge Unacceptable(x)\big) \Big) \quad ((4.5a)_{\text{HOEBAF}})$$

$$\forall x \in (Arg \cup Att \cup Sup)\big(Unacceptable(x) \rightarrow \neg Acceptable(x) \big) \quad ((4.5b)_{\text{HOEBAF}})$$

Support

$$\forall x \in (Arg \cup Att \cup Sup)\Bigg(\Big[PrimaFacie(x) \vee \exists \alpha \in Sup\Big(T(\alpha, x) \wedge \forall a \in Arg\big(S(\alpha, a) \rightarrow Selected(a) \big) \wedge Selected(\alpha) \Big)\Big] \rightarrow Supported(x) \Bigg) \quad ((4.6)_{\text{HOEBAF}})$$

Self-support

$$\forall x \in (Arg \cup Att \cup Sup)\Bigg(Supported(x) \rightarrow \Big[PrimaFacie(x) \vee \exists \alpha \in Sup\big(T(\alpha, x) \wedge \forall a \in Arg\big(S(\alpha, a) \rightarrow Selected(a) \big) \wedge Selected(\alpha) \big)\Big] \Bigg) \tag{(4.7a)$_{\text{HOEBAF}}$}$$

$$\forall x \in (Arg \cup Att \cup Sup)\Bigg(Unsupportable(x) \leftrightarrow \Big[\neg PrimaFacie(x) \wedge \forall \alpha \in Sup\big(T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(S(\alpha, a) \wedge Unsupportable(a) \big) \vee Unsupportable(\alpha) \vee \exists \beta \in Att\Big(\big[T(\beta, \alpha) \vee \exists a \in Arg(S(\alpha, a) \wedge T(\beta, a))\big] \wedge \big[\forall b \in Arg\big(S(\beta, b) \rightarrow Selected(b) \big) \wedge Selected(\beta)\big]\Big)\Big] \big)\Big]\Bigg) \tag{(4.7b)$_{\text{HOEBAF}}$}$$

Defence

$$\forall x \in (Arg \cup Att \cup Sup)\Bigg(Acceptable(x) \rightarrow \forall \alpha \in Att\Big(T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(S(\alpha, a) \wedge Unsupportable(a) \big) \vee Unsupportable(\alpha) \vee \exists \beta \in Att\Big(\big[T(\beta, \alpha) \vee \exists a \in Arg(S(\alpha, a) \wedge T(\beta, a))\big] \wedge \big[\forall b \in Arg\big(S(\beta, b) \rightarrow Selected(b) \big) \wedge Selected(\beta)\big]\Big)\Big] \Big)\Bigg) \tag{(4.8)$_{\text{HOEBAF}}$}$$

Reinstatement

$$\forall x \in (Arg \cup Att \cup Sup)\Bigg(\forall \alpha \in Att\Big(T(\alpha, x) \rightarrow \Big[\exists a \in Arg\big(S(\alpha, a) \wedge Unsupportable(a) \big) \vee Unsupportable(\alpha) \vee \exists \beta \in Att\Big(\big[T(\beta, \alpha) \vee \exists a \in Arg(S(\alpha, a) \wedge T(\beta, a))\big] \wedge \big[\forall b \in Arg\big(S(\beta, b) \rightarrow Selected(b) \big) \wedge Selected(\beta)\big]\Big)\Big] \Big) \rightarrow Acceptable(x) \Bigg) \tag{(4.9)$_{\text{HOEBAF}}$}$$

Complement attack

$$\forall x \in (Arg \cup Att \cup Sup)\Big(\neg Acceptable(x) \rightarrow \exists \alpha \in Att\big(T(\alpha, x) \wedge \forall a \in Arg(\ S(\alpha, a) \rightarrow Selected(a)\ ) \wedge Selected(\alpha)\ \big)\Big) \qquad ((4.10a)_{\text{HOEBAF}})$$

$$\forall x \in (Arg \cup Att \cup Sup)\big(\neg Supported(x) \rightarrow Unsupportable(x)\ \big) \qquad ((4.10b)_{\text{HOEBAF}})$$

Note. Again, Formula ($4.5a_{\text{HOEBAF}}$) can be rewritten in the same way as Formula ($4.5a_{\text{AF}}$) in Section 4.5.3.

Properties 1.1 to 1.6 hold whenever $\mathcal{A}$ is a Higher-Order Evidence-Based Argumentation Framework using formulas (4.17), as well as Axioms 4.11 and (4.12) as Σ'. This is captured by the following proposition.

Proposition 14. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an HO-EBAF and $U = (S, \Gamma, \Delta)$ be a structure.*

1. *U is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
2. *U is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
3. *U is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
4. *U is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
5. *U is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
6. *U is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*

4.6 Summary

In this section, we summarize the contribution of the present chapter. We provide several tables and figures that succinctly present our main results. At the end of the section, we present a recap example of how we envision our logical encoding to be used.

4.6.1 Logical Encoding

Table 4.4 summarizes the previous sections by giving for each Abstract Argumentation Framework the list of formulas that corresponds to its encoding. We remind the reader that in the present work, we are only interested in the evidential interpretation of the support relation without support cycle, and the RAF interpretation of higher-order interactions.

An important point is the fact that, using the hierarchy between frameworks, our generic approach gives the way to logically represent the Abstract Argumentation Frameworks which have not been directly addressed in the present work (i.e. HO-AF-C, EBAF-C, HO-EBAF-C), as indicated in Table 4.5.

Figure 4.2 presents the Abstract Argumentation Frameworks that can be taken into account with our approach. Please notice that Figure 4.2 corresponds to Figure 4.1 in which the support relation is interpreted as evidential support without support cycles.

	Encoding for		Specialisations	Parameters
	Graph	Semantics		
Generic rep.	(4.1) to (4.3),	(4.4) to (4.10)		
AF		(4.4), $(4.5)_{\text{AF}}$, $(4.8)_{\text{AF}}$ to $(4.10a)_{\text{AF}}$	(4.11) to (4.12)	(4.13)
AF-C		(4.4), $(4.5)_{\text{AF}}$, $(4.8)_{\text{AF}}$ to $(4.10a)_{\text{AF}}$	(4.12)	(4.13)
HO-AF		(4.4), $(4.5)_{\text{HOAF}}$ $(4.8)_{\text{HOAF}}$ to $(4.10a)_{\text{HOAF}}$	(4.11) to (4.12)	(4.15)
EBAF		(4.4), $(4.5)_{\text{EBAF}}$ to $(4.10)_{\text{EBAF}}$		(4.16)
HO-EBAF		(4.4), $(4.5)_{\text{HOEBAF}}$ to $(4.10)_{\text{HOEBAF}}$		(4.17)

Table 4.4: List of formulas for the encoding of each Abstract Argumentation Framework mentioned in Section 4.5

	Encoding for		Specialisations	Parameters
	Graph	Semantics		
Generic rep.	(4.1) to (4.3),	(4.4) to (4.10)		
HO-AF-C		(4.4), $(4.5)_{\text{HOAF}}$, $(4.8)_{\text{HOAF}}$ to $(4.10a)_{\text{HOAF}}$	(4.12)	(4.15)
EBAF-C		(4.4), $(4.5)_{\text{EBAF}}$ to $(4.10)_{\text{EBAF}}$		(4.16)
HO-EBAF-C		(4.4), $(4.5)_{\text{HOEBAF}}$ to $(4.10)_{\text{HOEBAF}}$		(4.17)

Table 4.5: List of formulas for the encoding of additional Abstract Argumentation Frameworks from those mentioned in Section 4.5 adding the Coalition enrichment

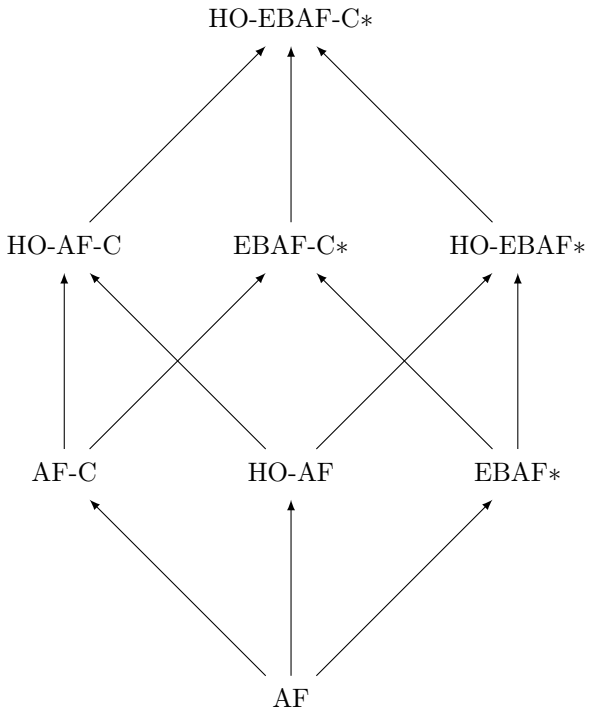

Figure 4.2: Hierarchy of classes of Abstract Argumentation Frameworks taken into account by our generic approach (∗: without support cycles)

4.6.2 Recap Example

We now present an example to illustrate how our logical encoding works. To this end, we extend the example presented in Section 3.6.2. To briefly recall its context, we suppose that a political debate is held between two candidates, Candidate 1 and Candidate 2, and we aim at identifying which of their propositions can form a satisfying conclusion to the debate. Thus, a computer program has been used to model the debate with an Argumentation Framework, depicted on Figure 3.42, and it has been decided that a satisfying conclusion would correspond to a stable extension of that Argumentation Framework.

Now, we know from Section 3.6.2 that such a stable extension is $\{b, c, g, i, l, p\}$. The question we are interested in, and which we eluded in Section 3.6.2, is how such an extension can be obtained. There are of course many ways to do so, but for the sake of the example, we will use our logical encoding.

So, we aim at showing how our logical encoding can compute the stable extension $\{b, c, g, i, l, p\}$ in the Argumentation Framework of Figure 3.42. Actually, we will use a version of this Argumentation Framework in which the arcs have received labels, as pictured on Figure 4.3. The link between the models our logical encoding computes and the extensions of an Argumentation Framework is formalised in Property 1, and more specifically, Property 1.6 in the case of stable extensions. Moreover, since we are computing an extension of an Argumentation Framework (i.e. without enrichments), our logical encoding must instantiated as shown in Section 4.5.3. Thus, the result we will use is in fact Proposition 10[4], and more specifically, Proposition 10.6 in the case of stable extensions.

What Proposition 10.6 essentially says is that, to compute a stable extension, like $\{b, c, g, i, l, p\}$, we must compute a Herbrand model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ (with $\mathcal{A}$ the Argumentation Framework of Figure 3.42) in which the predicate *Selected* is true for b, c, g, i, l and p (and only them). Recall that

[4]Recall that Proposition 10 is in fact just an instantiating of Property 1 for Argumentation Frameworks.

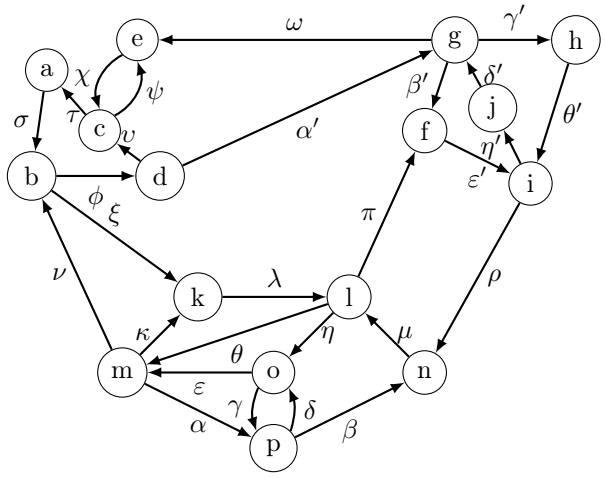

Figure 4.3: The Argumentation Framework of Figure 3.42 with labelled arcs

$\Sigma_{CA}(\mathcal{A})$ is made of Axioms (4.1), (4.2), (4.3) and of formulas (4.4), (4.5), (4.6), (4.7), and (4.10). Since here we compute an extension of an Argumentation Framework, our model must satisfy formulas (4.13). This means that formulas (4.5) and (4.10a) are replaced by formulas $(4.5)_{\text{AF}}$ and $(4.10a)_{\text{AF}}$. It also means that Propositions 6 to 9 can be used to simplify the effects of formulas (4.6), (4.7), and (4.10b).

This being said, we can start to see how these formulas affect our models. We start with Axioms (4.1) and (4.2). They give the value of the predicates *Arg*, *Att* and *Sup* for the elements of the domain. Axiom (4.3) can be used to obtain values for the predicates S and T. These three axioms together basically encode the structure of the Argumentation Framework at hand. Additionally, because our Argumentation Framework does not have a support relation, we know that the predicate *PrimaFacie* is going to be true for every element, and by formula (4.13a) we can also get the value of the predicate *Cand* for every element. Finally, Propositions 6 and 8 give us some values for the predicates *Supported* and *Unsupportable*. All of this is grouped in Tables 4.6 and 4.7 for the unary predicates, Tables 4.8 and 4.9 for the binary predicate S, and Tables 4.10 and 4.11 for the binary predicate T.

We can observe that Tables 4.9 and 4.11 are completely empty, so we could have chosen not to display them. Instead, we chose to do it anyway for the sake of being exhaustive. Additionally, since our models must satisfy Axioms (4.11) and (4.12) (because we are working with an Argumentation Framework without enrichments), in Tables 4.8 to 4.11, on each line where a "✓" is present, all other boxes of the line should have an "X". We chose not to display them this way for readability reasons (the "✓" would have been way more difficult to spot).

Now, all the possible models must start from the basis displayed on Tables 4.6 to 4.11. Starting from there, we can decide the truth value of other predicates, and the rest of the formulas our models must satisfy will propagate the consequences on other predicates. Of course, here, the "decision variables" so to say are the values of the predicate *Selected* for each element, which eventually encode the extension being computed. So we can decide of the value of this predicate for an element, for instance $Selected(b) = \top$. From this, we get $Acceptable(b) = \top$ using formula (4.4). Using formula (4.13b), since $S(\phi, b) = \top$, we get $Activable(\phi) = \top$, which leads to $Defeated(d) = \top$ using formula (4.13c) and the fact that $T(\phi, d) = \top$, and thus to $Inhibited(\upsilon) = \top$, $Inhibited(\alpha') = \top$, $Desactivated(\upsilon) = \top$ and $Desactivated(\alpha') = \top$ using formulas (4.13d) and (4.13e) because $S(\upsilon, d) = \top$ and $S(\alpha', d) = \top$. We have analogous results for ξ, k and λ, because $S(\xi, b) = \top$, $T(\xi, k) = \top$ and $S(\lambda, k) = \top$. In addition, by using formula (4.5), we obtain $Unacceptable(d) = \top$ and $Unacceptable(k) = \top$, which then leads to $Acceptable(d) = \bot$ and $Acceptable(k) = \bot$. Using formula (4.4) again, we thus get $Selected(d) = \bot$ and $Selected(k) = \bot$. Finally, by contrapositive reasoning, formula (4.5) gives us $Unacceptable(b) = \bot$ which, because b is the only element such that

	Arg	*Att*	*Sup*	*Cand*	*PrimaFacie*	*Selected*	*Acceptable*	*Unacceptable*
a	✓	X	X	✓	✓			
b	✓	X	X	✓	✓			
c	✓	X	X	✓	✓			
d	✓	X	X	✓	✓			
e	✓	X	X	✓	✓			
f	✓	X	X	✓	✓			
g	✓	X	X	✓	✓			
h	✓	X	X	✓	✓			
i	✓	X	X	✓	✓			
j	✓	X	X	✓	✓			
k	✓	X	X	✓	✓			
l	✓	X	X	✓	✓			
m	✓	X	X	✓	✓			
n	✓	X	X	✓	✓			
o	✓	X	X	✓	✓			
p	✓	X	X	✓	✓			
α	X	✓	X	X	✓			
β	X	✓	X	X	✓			
γ	X	✓	X	X	✓			
δ	X	✓	X	X	✓			
ε	X	✓	X	X	✓			
η	X	✓	X	X	✓			
θ	X	✓	X	X	✓			
κ	X	✓	X	X	✓			
λ	X	✓	X	X	✓			
μ	X	✓	X	X	✓			
ν	X	✓	X	X	✓			
ξ	X	✓	X	X	✓			
π	X	✓	X	X	✓			
ρ	X	✓	X	X	✓			
σ	X	✓	X	X	✓			
τ	X	✓	X	X	✓			
υ	X	✓	X	X	✓			
ϕ	X	✓	X	X	✓			
χ	X	✓	X	X	✓			
ψ	X	✓	X	X	✓			
ω	X	✓	X	X	✓			
α'	X	✓	X	X	✓			
β'	X	✓	X	X	✓			
γ'	X	✓	X	X	✓			
δ'	X	✓	X	X	✓			
ε'	X	✓	X	X	✓			
η'	X	✓	X	X	✓			
θ'	X	✓	X	X	✓			

Table 4.6: Starting point of a model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.13)\}$ for the Argumentation Framework of Figure 4.3 using formulas (4.1), (4.2), (4.6), (4.7), (4.13)
(✓: True, X: False)

	Activable	*Defeated*	*Inhibited*	*Desactivated*	*Supported*	*Unsupportable*
a					✓	X
b					✓	X
c					✓	X
d					✓	X
e					✓	X
f					✓	X
g					✓	X
h					✓	X
i					✓	X
j					✓	X
k					✓	X
l					✓	X
m					✓	X
n					✓	X
o					✓	X
p					✓	X
α						
β						
γ						
δ						
ε						
η						
θ						
κ						
λ						
μ						
ν						
ξ						
π						
ρ						
σ						
τ						
υ						
ϕ						
χ						
ψ						
ω						
α'						
β'						
γ'						
δ'						
ε'						
η'						
θ'						

Table 4.7: Starting point of a model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.13)\}$ for the Argumentation Framework of Figure 4.3 using formulas (4.1), (4.2), (4.6), (4.7), (4.13) (Continued)

$\boldsymbol{S}$	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	α	β	γ	δ	ε	η
a																						
b																						
c																						
d																						
e																						
f																						
g																						
h																						
i																						
j																						
k																						
l																						
m																						
n																						
o																						
p																						
α													✓									
β																✓						
γ															✓							
δ																✓						
ε															✓							
η												✓										
θ												✓										
κ													✓									
λ											✓											
μ														✓								
ν													✓									
ξ		✓																				
π												✓										
ρ									✓													
σ	✓																					
τ			✓																			
υ				✓																		
ϕ		✓																				
χ					✓																	
ψ			✓																			
ω							✓															
α'				✓																		
β'							✓															
γ'							✓															
δ'										✓												
ε'						✓																
η'									✓													
θ'								✓														

Table 4.8: Model of formula (4.3), for the Argumentation Framework of Figure 4.3 regarding the S predicate (✓: True, X: False)
(Empty cells on the same line as a "✓" should have an "X" to respect Axiom (4.11))

$\boldsymbol{S}$	θ	κ	λ	μ	ν	ξ	π	ρ	σ	τ	υ	ϕ	χ	ψ	ω	α'	β'	γ'	δ'	ε'	η'	θ'
a																						
b																						
c																						
d																						
e																						
f																						
g																						
h																						
i																						
j																						
k																						
l																						
m																						
n																						
o																						
p																						
α																						
β																						
γ																						
δ																						
ε																						
η																						
θ																						
κ																						
λ																						
μ																						
ν																						
ξ																						
π																						
ρ																						
σ																						
τ																						
υ																						
ϕ																						
χ																						
ψ																						
ω																						
α'																						
β'																						
γ'																						
δ'																						
ε'																						
η'																						
θ'																						

Table 4.9: Model of formula (4.3), for the Argumentation Framework of Figure 4.3 regarding the S predicate (Continued)
(Empty cells on the same line as a "✓" should have an "X" to respect Axiom (4.11))

$\boldsymbol{T}$	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	α	β	γ	δ	ε	η
a																						
b																						
c																						
d																						
e																						
f																						
g																						
h																						
i																						
j																						
k																						
l																						
m																						
n																						
o																						
p																						
α																✓						
β														✓								
γ																✓						
δ															✓							
ε													✓									
η															✓							
θ													✓									
κ										✓												
λ												✓										
μ												✓										
ν		✓																				
ξ											✓											
π						✓																
ρ														✓								
σ		✓																				
τ	✓																					
υ			✓																			
ϕ				✓																		
χ			✓																			
ψ					✓																	
ω					✓																	
α'							✓															
β'						✓																
γ'								✓														
δ'							✓															
ε'									✓													
η'										✓												
θ'									✓													

Table 4.10: Model of formula (4.3), for the Argumentation Framework of Figure 4.3 regarding the T predicate (✓: True, X: False)
(Empty cells on the same line as a "✓" should have an "X" to respect Axiom (4.12))

$\boldsymbol{T}$	θ	κ	λ	μ	ν	ξ	π	ρ	σ	τ	υ	ϕ	χ	ψ	ω	α'	β'	γ'	δ'	ε'	η'	θ'
a																						
b																						
c																						
d																						
e																						
f																						
g																						
h																						
i																						
j																						
k																						
l																						
m																						
n																						
o																						
p																						
α																						
β																						
γ																						
δ																						
ε																						
η																						
θ																						
κ																						
λ																						
μ																						
ν																						
ξ																						
π																						
ρ																						
σ																						
τ																						
υ																						
ϕ																						
χ																						
ψ																						
ω																						
α'																						
β'																						
γ'																						
δ'																						
ε'																						
η'																						
θ'																						

Table 4.11: Model of formula (4.3), for the Argumentation Framework of Figure 4.3 regarding the T predicate (Continued)
(Empty cells on the same line as a "✓" should have an "X" to respect Axiom (4.12))

$T(\sigma, b) = \top$ and $T(\nu, b) = \top$, leads to $Activable(\sigma) = \bot$ and $Activable(\nu) = \bot$. Then, using formula (4.13b), we have $Selected(a) = \bot$ and $Selected(m) = \bot$, which leads to $Acceptable(a) = \bot$ and $Acceptable(m) = \bot$.

At this stage, if we only wanted to compute a conflict-free extension, we would be done with propagating values in the model. Please remark that, in this case, our extension is not fully encoded since the value of the predicate *Selected* is not known for all the elements that have $\top$ as the value for the predicate *Cand*. However, having stopped propagating values means that we have reached a somewhat "stable" situation (not to be confused with the notion of stable extension) in the sense that we could decide to stop there, put $\bot$ as the value of *Selected* for all the other elements with $\top$ as their value for *Cand* and that do not have a value for *Selected*, propagate the necessary values according to formulas, and obtain a correct model. The model would then encode the extension $\{b\}$ which is a correct conflict-free extension. Alternatively, we could select another element than b with the value $\top$ for the predicate *Cand* and no value for the predicate *Selected*, like c, and decide that $Selected(c) = \top$. we would then need to do another round of value propagation, and so on.

However, since our models must satisfy formula (4.10a), here it follows from $Acceptable(a) = \bot$ and $Acceptable(m) = \bot$ that $Defeated(a) = \top$ and $Defeated(m) = \top$. Without detailing all the consequences of these new values, they lead in particular to $Activable(\tau) - \top$ (for a) and either $Activable(\theta) = \top$ or $Activable(\varepsilon) = \top$ (for m). Thus, we have $Selected(c) = \top$ and either $Selected(l) = \top$ or $Selected(o) = \top$. Suppose that we decide to have $Selected(l) = \top$. Then, $Selected(c) = \top$ and $Selected(l) = \top$ would lead to other rounds of value propagation similar to the one we had with b which we will not detail. In the end, we obtain the model described by Tables 4.12 and 4.13 (still using Tables 4.8 to 4.11 for the binary predicates). The extension encoded by this model is indeed $\{b, c, g, i, l, p\}$.

As a finishing note, we can observe that Tables 4.8 to 4.13 which display our final model are incomplete, in the sense that some predicates are not assigned a value for some elements. This is because the value of those predicates for those specific elements is in fact irrelevant, and does not impact the rest of the model. We leave it to the reader to verify it. This means in particular that, instead of letting their value open, we could simply force a value for them, which would greatly limit the time needed to compute a model. Please observe as well that regarding the predicate *Selected*, in our case, we could assign the value $\top$ to elements like $\alpha, \beta, \ldots$ which have the value $\bot$ for the predicate *Cand* and it would indeed not be contradictory with the theory. However, in virtue of Proposition 10.6, only the models that assign the value $\bot$ to those elements are used to compute extensions.

4.7 Related works

First of all, it is interesting to note that the idea of defining a general account of Abstract Argumentation Frameworks has been introduced in [BGW05] under the name of "higher-level networks" and pursued by Gabbay through several papers [Gab09a, Gab09b], using the idea of meta-argumentation. In these networks, one can find interactions (attacks or supports) between interactions at any level; moreover, interactions can be collective (the source is a set of elements) or disjunctive (the target is a set of elements); and finally, each element of these networks can be valued. Gabbay's aim was to define a framework rich enough to generalize all the existing networks. Nevertheless, due to this richness of representation, only informal examples are given and most formal definitions are missing. For instance, the meaning of support is left unknown and the only existing semantics are defined for a version of this network restricted to higher-order attacks that are not collective (see [Gab09a, Gab09b]).

Consider now the approach proposed by Cayrol et al. in [CL20] that proposed an encoding of AF, RAF and REBAF (HO-EBAF in the present work) and their semantics into first-order logics with finite domains. Their approach is however not generic. In the present work, we clearly extend it through our generic representation, taking into account AF, HO-AF and HO-EBAF (as in [CL20]) but also EBAF and the collective versions of all these frameworks (AF-C, EBAF-C, HO-AF-C and HO-EBAF-C).

The Abstract Dialectical Frameworks (ADF) is another interesting approach, introduced many years ago (see [BES+18] for a recent synthetic paper about ADF). This approach is related to ours in the sense that

	Arg	*Att*	*Sup*	*Cand*	*PrimaFacie*	*Selected*	*Acceptable*	*Unacceptable*
a	✓	X	X	✓	✓	X	X	✓
b	✓	X	X	✓	✓	✓	✓	X
c	✓	X	X	✓	✓	✓	✓	X
d	✓	X	X	✓	✓	X	X	✓
e	✓	X	X	✓	✓	X	X	✓
f	✓	X	X	✓	✓	X	X	✓
g	✓	X	X	✓	✓	✓	✓	X
h	✓	X	X	✓	✓	X	X	✓
i	✓	X	X	✓	✓	✓	✓	X
j	✓	X	X	✓	✓	X	X	✓
k	✓	X	X	✓	✓	X	X	✓
l	✓	X	X	✓	✓	✓	✓	X
m	✓	X	X	✓	✓	X	X	✓
n	✓	X	X	✓	✓	X	X	✓
o	✓	X	X	✓	✓	X	X	✓
p	✓	X	X	✓	✓	✓	✓	X
α	X	✓	X	X	✓			
β	X	✓	X	X	✓			
γ	X	✓	X	X	✓			
δ	X	✓	X	X	✓			
ε	X	✓	X	X	✓			
η	X	✓	X	X	✓			
θ	X	✓	X	X	✓			
κ	X	✓	X	X	✓			
λ	X	✓	X	X	✓			
μ	X	✓	X	X	✓			
ν	X	✓	X	X	✓			
ξ	X	✓	X	X	✓			
π	X	✓	X	X	✓			
ρ	X	✓	X	X	✓			
σ	X	✓	X	X	✓			
τ	X	✓	X	X	✓			
υ	X	✓	X	X	✓			
ϕ	X	✓	X	X	✓			
χ	X	✓	X	X	✓			
ψ	X	✓	X	X	✓			
ω	X	✓	X	X	✓			
α'	X	✓	X	X	✓			
β'	X	✓	X	X	✓			
γ'	X	✓	X	X	✓			
δ'	X	✓	X	X	✓			
ε'	X	✓	X	X	✓			
η'	X	✓	X	X	✓			
θ'	X	✓	X	X	✓			

Table 4.12: Model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.13)\}$ for the Argumentation Framework of Figure 4.3
(✓: True, X: False)
(Empty cells could have either a "✓" or an "X" without changing the values already displayed)

	Activable	*Defeated*	*Inhibited*	*Desactivated*	*Supported*	*Unsupportable*
a		✓			✓	X
b		X			✓	X
c		X			✓	X
d		✓			✓	X
e		✓			✓	X
f		✓			✓	X
g		X			✓	X
h		✓			✓	X
i		X			✓	X
j		✓			✓	X
k		✓			✓	X
l		X			✓	X
m		✓			✓	X
n		✓			✓	X
o		✓			✓	X
p		X			✓	X
α	X		✓	✓		
β	✓		X	X		
γ	X		✓	✓		
δ	✓		X	X		
ε	X		✓	✓		
η	✓		X	X		
θ	✓		X	X		
κ	X		✓	✓		
λ	X		✓	✓		
μ	X		✓	✓		
ν	X		✓	✓		
ξ	✓		X	X		
π	✓		X	X		
ρ	✓		X	X		
σ	X		✓	✓		
τ	✓		X	X		
υ	X		✓	✓		
ϕ	✓		X	X		
χ	X		✓	✓		
ψ	✓		X	X		
ω	✓		X	X		
α'	X		✓	✓		
β'	✓		X	X		
γ'	✓		X	X		
δ'	X		✓	✓		
ε'	X		✓	✓		
η'	✓		X	X		
θ'	X		✓	✓		

Table 4.13: Model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.13)\}$ for the Argumentation Framework of Figure 4.3 (Continued)
(Empty cells could have either a "✓" or an "X" without changing the values already displayed)

it proposes an encoding of AF, BAF (and even HO-AF by a flattening process) and their collective versions using logics. Nevertheless the mechanisms used in the ADFs are completely different from what we develop. Indeed, the ADF input consists of a dependence graph (nodes are arguments, statements or positions and edges are dependence links) together with an acceptance condition attached to each node. This condition can be (and generally is) a propositional formula expressing the way the status of the argument is impacted by the status of its parents in the graph. Then the computation of the labellings or extensions is done using the models of this dependence graph. It thus allows a great liberty as to how to decide of the status of an argument, capturing a lot of different frameworks. In the present work, genericity lies in the use of a common group of general formulas for each framework. These formulas are then specialized to correspond to a given framework based on the enrichments (or combinations thereof) that are present or not.

Another related work is proposed in [GG15]. The authors also propose a logical encoding of an Argumentation Framework using propositional logic. It is worth noticing that this encoding is quite similar to the one proposed in [CL20]. And once again, the computation of semantics is done through the logical models of the proposed theory. Nevertheless, this approach is only defined for Argumentation Frameworks, and, as it stands, fails to be generic like the one in the present work is.

A more specific work that also proposed a logical encoding of an Argumentation Framework and a mechanism for computing its extensions is described in [dSBCL16]. This approach is not generic as well. Nevertheless it has the advantage of being usable for expressing the properties of update operators in dynamic argumentation. Moreover, it enables to express incomplete knowledge about an Argumentation Framework, and so to describe a set of Argumentation Frameworks by a single formula (each model of this formula corresponding to a particular Argumentation Framework).

A more recent approach described in [AGPT20, AGPT21] also proposed an encoding of some enriched frameworks. This encoding is given under the form of a logic program, each element of the framework producing a specific rule characterising the conditions for belonging to a complete extension/structure. Then a correspondence between complete extensions/structures and some special models (partial stable models) of the logic program is given. Unlike us, this approach does not rely on the underlying principles of each semantics (Coherence, Defence, Reinstatement, ...). Moreover, the scope of this work does not exactly match ours: the logics that are used are clearly different; we take into account more semantics; the studied enrichments are different (coalitions, higher-order relations with the RAF interpretation and evidential supports for us, and higher-order relations with the RAF or AFRA interpretation and necessary or deductive supports for them). Nevertheless, there clearly are some common points between their approach and the present work.

Finally, there exists some other approaches proposing an encoding of an Argumentation Framework (or its enrichments) and a characterization of its extensions either by some specific models of a given logic program or by the models of some specific logic programs (see for instance [EGW10, CNO09, ON17], the last reference giving a summary of the numerous existing characterizations). Nevertheless, none of these works propose a generic representation able to take into account indifferently an Argumentation Framework or one of its enriched version. We can also mention the work of [SF21] which proposes a representation of Abstract Argumentation in Henkin's Extensional Type Theory. This formalism, which is more expressive than First-Order Logic, allows the authors not only to assess Argumentation Frameworks under argumentation semantics, but also to explore meta-theoretical properties of argumentation.

4.8 Future Perspectives

There are several lines of future research that can be drawn for our generic logical encoding of Abstract Argumentation semantics. First of all, as we mentioned a few times, when both the support relation (interpreted as evidential support) and higher-order relations (with the RAF interpretation) are present, we consider for the moment that there is no support cycle. This is certainly a limitation that we ought to remove in order to have a logical encoding in the general case. The recent work of [Lag21] has solved this problem for a different

encoding, which we proved was captured by ours. Thus, one could think of integrating similar modifications to achieve the same results in our logical encoding.

A second line of research would be of course to extend our logical encoding so that more aspects of Abstract Argumentation are taken into account. This includes the integration of new enrichments, to capture more generalised Abstract Argumentation Frameworks, or even the integration of new Abstract Argumentation principles, to capture new semantics. However, on a closer perspective, this also includes the integration of the interpretations of the enrichments we already consider and that are not part of the encoding yet. This covers the AFRA interpretation of Higher-Order relations, the deductive and necessary interpretation of the support relation, taking into account coalitions of arguments as targets, and the missing interpretation for coalitions in both cases of source and targets (a relation is effective if at least one argument respects a property in its source, and a relation affects all or at least one argument of its target).

Now, we believe that our logical encoding is defined so that the integration of the different interpretations of coalitions should not be too much difficult to include. We assumed through the entire chapter that relations could only target a single arguments, which resulted in the inclusion of Axiom (4.12) in all the theories we considered. To have a theory taking sets of arguments as targets, it suffices to remove this axiom from the appropriate theory. Similarly, to move from a universal interpretation (all the arguments of the source/target) to an existential one (at least an argument of the source/target), we believe it suffices to change the quantifier ($\forall$ or $\exists$) into its dual counterpart at some specific places in the formulas. For instance, changing the $\forall a \in Arg$ by $\exists a \in Arg$ in the instantiating formula of the parameter predicate *Activable* would be one of the possibly several changes needed to have a theory corresponding to a framework which uses an existential interpretation of sets of arguments as sources of relations.

We have also already made some preliminary thoughts as to how to include the AFRA interpretation of higher-order relations into our logical encoding. The RAF and AFRA interpretations are very close but still separated by subtle differences. One obvious way to solve semantics for higher-order frameworks with the AFRA interpretation would be to work with the flattened Argumentation Framework (that is already included in the logical encoding) resulting from the higher-order one. However, we find that not to be very insightful. More problematic, we also believe that it could be an obstacle to combining higher-order relations with the AFRA interpretation with other enrichments. So, instead, we believe that the direct inclusion of this interpretation into the formulas of the encoding (certainly into the instantiating formulas since it could introduce a choice on whether to use the RAF or the AFRA interpretation) is more productive. Now, the obvious technical obstacle is that our theory fails to involve a second type of defeat (namely, the indirect defeat that is used in the AFRA interpretation). Despite this observation, a hint to deal with this problem would be to introduce two extra parameters:

- The first parameter, called $Challenged$, would express what type of defeat must be taken into account so as to capture conflict-freeness in the following way:

$$\forall x \in Cand(Challenged(x) \rightarrow Unacceptable(x)) \tag{4.18}$$

  with $Challenged(x)$ defined by:

  - $Defeated(x)$ for the RAF encoding
  - $Inhibited(x)$ for the AFRA encoding.

- The second parameter, called $ImpactedBy$, would express which attack can impact which item, so that defense would be written as follows:

$$\forall x \in Cand(Acceptable(x) \rightarrow \forall \alpha \in Att(ImpactedBy(x, \alpha) \rightarrow Desactivated(\alpha))) \tag{4.19}$$

  with $ImpactedBy(x, \alpha)$ defined by:

  - $T(\alpha, x)$ for the RAF encoding (x is impacted by α iff x is the target of α)

– $T(\alpha, x) \vee \exists a \in Arg(S(x, a) \wedge T(\alpha, a))$ for the AFRA encoding (x is impacted by α iff either x is the target of α or the target of α is part of the source of x).

These, of course, represent only preliminary thoughts and should be further studied to see if their intuition is correct. Finally, we also thought about how to integrate the deductive and necessary interpretation of the support relation. For now, our intuition is that these interpretation are too different from the evidential interpretation to have a set of generic formulas being able to capture them all, even with the extra flexibility introduced with the use of parameter predicates to instantiate. We thus believe that a different encoding would be necessary to capture these other interpretations of the support relation. However, as it is well known that deductive and necessary supports are dual, we also believe that the same encoding could be used to capture both of them at once. A recent work, [Lag23], has explored the logical encoding of HO-AFN-C (called RAFN in the literature) and could serve as a starting point for the integration of the deductive and necessary interpretation of the support relation.

It is possible to put the work done in this chapter in perspective with the work presented in Chapter 3. Indeed, Chapter 3 presented explanations for Abstract Argumentation semantics in Argumentation Frameworks. But, Abstract Argumentation semantics are precisely the object of the logical encoding presented in this chapter. Not only that, but the work done in this chapter is not restricted to Argumentation Frameworks only. As such, the logical encoding presented in this chapter, and especially the way it handles generalisations of Abstract Argumentation, can be valuable for one of the future perspectives mentioned at the end of Chapter 3: the extension of explanations to generalised accounts of Abstract Argumentation.

Chapter 5

Extension of the Logical Encoding: Computation of Explanations for Extensions

In this chapter, we extend the logical encoding of Abstract Argumentation Frameworks presented in Chapter 4 to include the computation of explanations presented in Chapter 3. The main point here is to have the computation of extensions and the computation of their explanations encapsulated into the same logical theory. Recall that the explanations of Chapter 3 have only been defined for Argumentation Frameworks. By integrating them into the generic logical theory of Chapter 4, we open the way to generalise them to any Abstract Argumentation Framework of Figure 4.2. As such, the work in this chapter is in fact the pursue of one of the future perspectives evoked in Chapter 3. In addition, having the computation of explanations expressed in the same logical theory as the computation of what they are about could allow a deeper formal study and analysis of their properties.

The chapter is organised as follows: firstly, we motivate further the work of this chapter (Section 5.1). Then, in Section 5.2, using the definitions of the explanations from Chapter 3, we discuss how they can be integrated into the logical theory. In particular, we wish for the encoding of the explanations to follow the same spirit as the encoding of the extensions, in that we want them to be *generic* as well. That is to say, for explanations as for extensions, we will have generic formulas that will require to be instantiated using some parameters to retrieve each individual kind of explanations. Following this discussion and the observations that we will make, we will then give the proper generic formulas to use, as well as each individual instantiating for each kind of explanations (Section 5.3). We then present some formal results on this logical encoding of explanations (Section 5.4). Finally, we discuss future directions of research that are still open for this body of work (Section 5.6).

5.1 Motivation

In this section, we develop further our motivation for the work done in this chapter. As we said previously, the purpose of this chapter is to propose a logical encoding of the visual explanations presented in Chapter 3 that can be integrated into the logical theory of Abstract Argumentation Frameworks presented in Chapter 4. By doing so, our objectives are firstly to have a first stone from which it is possible to study and build the generalisations of our explanations to the generalisations of Argumentation Frameworks considered in Chapter 4, and secondly, to have the possibility to study the links between our explanations and what they explain from the same perspective, that is formal logic. The reader may recall that the first objective corresponds to a future perspective that was discussed in Chapter 3.

Our first objective comes from the general observation that both our explanations of Chapter 3 and the

logical encoding of Chapter 4 are largely based on the decomposition of Abstract Argumentation semantics into principles, discussed in Section 2.4. As such, it appears possible to us to merge both approaches. While the explanations of Chapter 3 have only been defined for the basic case of Argumentation Frameworks, the logical encoding of Chapter 4 is designed to generically capture several of their generalisations, as well as to be able to include more of these generalisations in a facilitated way. It is perfectly possible to imagine extending the explanations of Chapter 3 to various generalisations of Argumentation Frameworks separately and independently. This has in fact been discussed as a future line of research in Chapter 3, and thus constitutes a direct follow up of the work done in that chapter. Moreover, the logical encoding of Chapter 4 gives us precious insights as to how the presence/absence of some enrichment (i.e. generalisation) impacts the computation of semantics. Since the computation of extensions is *precisely* what the explanations of Chapter 3 are designed to explain, it would be a shame not to make use of these insights if we are to extend those explanations to the different generalisations of Argumentation Frameworks. Hence, integrating an encoding of our explanations into the generic logical theory of Chapter 4 should bring us one step closer to the definitions of generalised explanations. It should be noted that the definition of generalised explanations could allow us to identify the general principles that rule over our explanation process, principles that could be missed due to the fact that we only study explanations for Argumentation Frameworks at the moment. Moreover, it should also be noted that, since the logical encoding of Chapter 4 is thought to facilitate the integration of new generalisations, integrating the computation of explanations into this theory should also facilitate the adaptation of these explanations to the new generalisations that could be introduced in the future.

Regarding our second objective, the point is to have a better understanding of the nature of the explanations we defined in Chapter 3. To do that, one way could be to see if they correspond, or at least relate to some level, to other notions of explanations in different domains. This is precisely the idea here. Indeed, there are close ties between the study of explanations for AI systems and formal systems. Already back in the time where we wanted to explain expert systems and the likes, the relations between explanations and formal logic were being considered relevant. This is not surprising as formal logic is an integral part of expert systems. Although there is no consensus on a particular definition of what an explanation or what the explanatory process is in formal logic, there are notions related to explanations that have rather satisfactory counterparts in formal logic. For instance, it is generally agreed upon that an explanation delivers causes, and that humans select these causes via an abductive reasoning process ([Har65]). An abductive reasoning process is basically trying to determine the correct causes of an event among a set of possible causes. This process can be modeled in formal logic through the notion of an abductive reasoning problem. Similarly, the formal logic notion of implicant ([Qui59]) is usually deemed a satisfactory representation of the notion of cause (and so, of explanation). The term "implicant" has different meanings depending on the context in which it is used. Thus, in general, an implicant simply is the hypothesis of an implication (so it is A in the formula $A \rightarrow B$). When referring to a boolean function, an implicant is a conjunction of litterals so that when the implicant is true, so is the function. In the last case, it is possible to have more specific notions, so for instance a prime implicant is an implicant that cannot be covered by a more reduced implicant. These were only a few examples, but the general point is: do our explanations coincide with some of these notions? Can they be obtained via an abductive reasoning problem from the encoding of Chapter 4? Are they implicants or even prime implicants of this encoding? These are examples of questions that, we believe, can bring precious insights on the nature of our explanations from a different perspective, and thus deserve to be investigated.

5.2 Identifying Shared Structures

In this section, we work towards identifying how to integrate the explanations defined in Chapter 3 into the logical theory of Chapter 4. Recall that an important point is to do so following the spirit of the generic logical theory. Thus, explanations should be encoded using some generic formulas that describe common ground on which rely the explanations, and then an instantiating of some parameters that, when combined with the generic formulas, allows to retrieve each individual kind of explanations.

Before we begin, we consider necessary to inform the reader that, despite our efforts, we could not find a way to have what we will call a *complete integration* of the explanations into the theory of Chapter 4. What we mean by that is that, for a given Argumentation Framework $\mathcal{A}$, we found no way to encode explanations that uses strictly the same language $\mathfrak{L}_{Ext}(\mathcal{A})$ as given in Section 4.5.1. More precisely, we found no way to make the predicates given in Chapter 4 interact so that a set of formulas that use them could indeed compute the desired explanations. Although, this would have probably been preferable to pursue our objectives of generalising explanations to enriched frameworks and study the relation between the computation of extensions and the computation of explanations form a logical perspective, this is maybe not much of a surprise. Indeed, this may simply indicate that our explanations are based on *different concepts* than those used to compute extensions. As such, the explanations are here encoded using a *different language* than the one used to compute the extensions that are to be explained.

However, we should also point out that this language is not *entirely different*, as there are still some common grounds with the language $\mathfrak{L}_{Ext}(\mathcal{A})$. In particular, the two languages share the same predicates, representing the same concepts, used to encode the *structure* (i.e. the *graph*) of the Argumentation Framework at hand, as well as the *extension* that is considered. As such, while still being different, the two languages used (one to compute extensions, one to compute explanations) would still have some shared elements, that are in fact the predicates that describe the graph and those that describe the extension.

That being said, we now move on to discussing how we can logically encode in a generic way the explanations of Chapter 3. Of course, such a generic encoding would heavily rely on the shared parts in the structure of these explanations. These shared structures are destined to become the generic formulas that are to be instantiated to retrieve each individual kind of explanations. Then, what is left would then be the individual parts of each kind of explanations, that are to become the instantiated parameters of the generic formulas through specific formulas. Thus, to identify these parts (shared and individual), we begin by recalling the definitions of each kind of explanations (the original definitions can be found in Chapter 3).

Definition. 34 Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, and consider $X = \{(a,b) \in \mathscr{R} \mid a,b \in S\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Coh} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = S$
- $\mathscr{R}' \subseteq X$
- If $X \neq \varnothing$, then $\mathscr{R}' \neq \varnothing$

Definition. 36 Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, and consider $X = \{(b,a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Def} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$
- $X \subseteq \mathscr{R}' \subseteq X \cup Y$
- $\forall b \in \mathscr{R}^{-1}(S)$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a,b) \in \mathscr{R}'$ with $a \in S$

Definition. 38 Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Rein1} for S on $\mathcal{A}$ if and only if

- $S \cap X \subseteq \mathscr{A}' \subseteq X$
- $\mathscr{R}' = \varnothing$
- If $(\mathscr{A} \setminus S) \cap X \neq \varnothing$, then $\exists a \in (\mathscr{A} \setminus S) \cap X$ with $a \in \mathscr{A}'$

Definition. 39 Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(b,c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q^{Ext}_{Rein2} for S on $\mathcal{A}$ if and only if

- $A' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$
- $X \subseteq \mathscr{R}' \subseteq X \cup Y$
- For every $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a, b) \in \mathscr{R}'$ with $a \in S$

Definition. 41 Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. A subgraph $(\mathscr{A}', \mathscr{R}')$ of $\mathcal{A}$ is an *answer to* Q_{CA}^{Ext} for S on $\mathcal{A}$ if and only if

- $\mathscr{A}' = \mathscr{A}$
- $\mathscr{R}' \subseteq X$
- $\forall b \in \mathscr{A} \setminus S$, if $b \in \mathscr{R}^{+1}(S)$, then $\exists (a, b) \in \mathscr{R}'$ with $a \in S$

It is immediately striking from the previous definitions that they indeed share a common structure. It may appear, though, that some definitions cannot be grouped with others. It is especially visible using the third condition of each definition. For some of the definitions (34 and 38), this condition is a general existential criterion (if ... then ...), while for the others (36, 39 and 41) it is a condition over all the arguments of a particular set. Notice further that in this case, the condition is always the same, i.e. if the argument of this particular set is attacked by the extension, then one of these attacks must belong to the explanation. In addition, for these explanations, the set of arguments is always the union of some subsets of arguments of the initial framework, and the set of attacks is always contained in the union of two subsets of attacks of the initial framework (think of the second set as being the empty set in the case of Definition 41).

As such, definitions 36, 39 and 41 seem sufficiently similar to be grouped together. It remains to see whether definitions 34 and 38 can also be grouped together. Although their third condition is similar in form, their sets of arguments and attacks behave quite differently from one another. However we can notice that the set of arguments of Definition 34 behaves as the set of attacks of Definition 38: plain equality with another set. Likewise, the set of attacks of Definition 34 behaves as the set of arguments of Definition 38: inclusion in another set and a part of that set being mandatory (think of the empty set for Definition 34). In addition, the third condition of Definition 34 is on attacks, while the third condition of Definition 38 is on arguments, i.e. elements that behave alike in the explanations. Thus, it seems possible to group definitions 34 and 38 together, by abstracting on which elements (arguments or attacks) the conditions are on.

The two groups of definitions we obtain might correspond to two different ways of explaining, each way being more appropriate for some semantics and not for the others. These ways of explaining may be better understood using the third condition of the definitions, which is the main support of the existence of two groups that cannot be further merged. Indeed, for the first group of definitions (definitions 34 and 38), the third condition is a *global* conditional existential criterion. In other terms, the fulfilment of some general condition at the level of the Argumentation Framework leads to the existence of some specific elements in the explanation. In addition, the general condition is usually the existence of such elements in the framework (i.e. a given set is not empty), thus making it easy to visually spot the information needed to interpret the explanation on it.

Example. Recall Figure 3.9, that shows an explanation on why $\{b, d, i\}$ does not respect the Coherence principle in Figure 2.3. The presence of merely one attack is enough to deduce that it is not conflict-free. This is why in Definition 34, if such attacks exist, we force at least one to appear, so that the explanation is interpreted correctly. Such a criterion can immediately be spotted on the picture.

On the other hand, for the second group of definitions (definitions 36, 39 and 41), the third condition is a conditional existential criterion on *every element of a given set*. That is to say, to interpret the explanation, one needs to spot a visual information on all the elements of an identified group (hence enforcing the existence of this information using the third condition). This is different from the first group of definitions because in this case the information is not global, and thus requires somewhat a closer look. Since the interpretation of the explanation requires the presence of information on all identified elements, one can also interpret it by looking for the absence of that information on one of the identified elements, thus making it simpler.

Example. Recall Figure 3.13, that shows the explanation of why $\{b, h, j\}$ does not respect the Defence principle in Figure 2.3. The presence of merely one source node among its attackers is enough to deduce so. This is why in Definition 36, we force the presence of at least one attack from the studied set to each of its attackers, if such an attack exists, so that the explanation is interpreted correctly. Such a criterion requires to inspect the attackers one after another, until one that is a source node has been found or all have been inspected. Yet, the property of being a source node is usually easily spotted.

5.3 A Family of Logical Theories for Explaining Abstract Argumentation

In this section, we use the discussion of the previous section to define several logical theories that aim at capturing the different explanations defined in Chapter 3. Much in the spirit of Chapter 4, the logical theories presented here are in fact all derived from a generic theory that can be instantiated through parameters.

5.3.1 A Generic Theory

As in Chapter 4, we begin with the generic part of the logical theory. Here as well, the formulas used are formulas of first order logic with equality. We also consider that the theory is relative to a given Argumentation Framework $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ and a given extension $S \subseteq \mathscr{A}$, which are taken for granted throughout. Importantly, to follow the spirit of Chapter 4, we in fact use its Higher-Order Evidence-Based Argumentation Framework notation $(\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ where $\mathscr{R}$ just contains the name of the attacks, $\mathscr{S}$ is empty, all the elements are prima-facie, and s and t are the usual source and target functions, with $s : \mathscr{R} \mapsto \mathscr{A}$ and $t : \mathscr{R} \mapsto \mathscr{A}$. We write $\mathfrak{L}_{Expl}(\mathcal{A})$ to denote the language of this theory.

Vocabulary

As in Chapter 4, we keep here the convention of using Latin letters to identify arguments and Greek letters to identify arcs (attacks only in this case). We also adopt the same convention regarding individual constants.

Individual Constants For all e in $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$, we take e to be an individual constant.

Thus again, the same letter designates both an element of the Argumentation Framework and the logical constant that represents it in $\mathfrak{L}_{Expl}(\mathcal{A})$.

Predicates The list of unary and binary predicates, as well as their intended meaning, is presented in Table 5.1.

First of all, please notice that this theory reuses the predicates $Arg, Att, Sup, PrimaFacie, Selected, S, T$ introduced in Chapter 4, with the same meanings. These represent the common grounds between the two languages $\mathfrak{L}_{Ext}(\mathcal{A})$ and $\mathfrak{L}_{Expl}(\mathcal{A})$.

The main new predicate that is introduced is $Expl$. $Expl(x)$ means that x belongs to the explanation (whether x is an argument or an attack). It will be this predicate that we use to characterize an explanation from a model of the theory. $ElemFixed(x)$ and $ElemVar(x)$ mean that x is a fixed and a variable element, respectively. What we call fixed elements, are the elements from the Argumentation Framework (either arguments or attacks) for which there is a clear and strict identification of those that belong to the explanation (represented by $ExplEF$). On the contrary, variable elements are the elements from the Argumentation Framework (either arguments or attacks) for which some must absolutely belong to the explanation and some others may or may not belong to it. We make the distinction between the two using the $NecessaryEV$ and $AdditionalEV$ predicates respectively. Finally, $ParticularEF$ and $ParticularEV$ represent some specific fixed and variable elements respectively.

The predicates $ElemFixed, ElemVar, ExplEF, ParticularEV, NecessaryEV, AdditionalEV$, and $ParticularEF$ are the parameters of this generic theory. They will be the predicates whose instantiating allows to retrieve each different kind of explanations.

	Unary Predicates	Meaning
	$Arg(x)$	x is an argument
	$Att(x)$	x is an attack
	$Sup(x)$	x is a support
	$PrimaFacie(x)$	x is a *prima facie* evidence
	$Selected(x)$	x is a member of the current extension
	$Expl(x)$	x is a member of the current explanation
✓	$ElemFixed(x)$	x is a fixed element
✓	$ElemVar(x)$	x is a variable element
✓	$ExplEF(x)$	x is a fixed element that is a member of the current explanation
✓	$ParticularEV(x)$	x is a particular variable element
✓	$NecessaryEV(x)$	x is a variable element that must belong to the explanation
✓	$AdditionalEV(x)$	x is a variable element that may belong to the explanation
✓	$ParticularEF(x)$	x is a particular fixed element

	Binary Predicates	Meaning
	$S(\alpha, x)$	x is in the source of α
	$T(\alpha, x)$	x is in the target of α

Table 5.1: Unary and Binary Predicates (✓ indicates those which are what we call parameters of the theory)

Axioms for the Argumentation Framework

As in Chapter 4, the purpose of the formulas given here is to encode the graph of the Argumentation Framework that is handled. We will use the same notation convention for the elements of $\mathcal{A}$ that was used in Chapter 4. With this, we can in fact use the very same Axioms that were presented back then. This is due to the fact that we precisely reuse the predicates that are relative to the encoding of the graph in this theory. As such, Axioms (4.1), (4.2), and (4.3) are part of this theory.

Note. Please note that, as a consequence, $\Sigma(\mathcal{A})$ is a valid notation in this theory as well.

Axioms for the Extension

Recall that the aim of the present theory is to compute explanations for a given extension. As we said previously, it is relative to a given Argumentation Framework, *and* to a given extension. Hence, we need some additional axioms to encode this extension. This is the role of the following formulas.

Axioms for the elements of the extension

$$\text{for all } a \in S, \quad Selected(a) \tag{5.1a}$$

$$\text{for all } a \notin S, \quad \neg Selected(a) \tag{5.1b}$$

Note. Keep in mind that here S designates our extension and should not be confused with our binary predicate encoding the source of the relations.

Notation. In the following, considering an AF $\mathcal{A}$ and a set of arguments S, we denote by $\Sigma(\mathcal{A}, S)$ the subtheory consisting of $\Sigma(\mathcal{A})$ and Axioms (5.1) together.

Generic Formulas for Explanations

The formulas given here aim at describing the explanations for the extension. As we mentioned earlier, we highlighted the existence of two different groups of explanations, that are different in that they rely on different ways of explaining. So, to be more precise, the formulas we give here encode the general structure of each of these two groups.

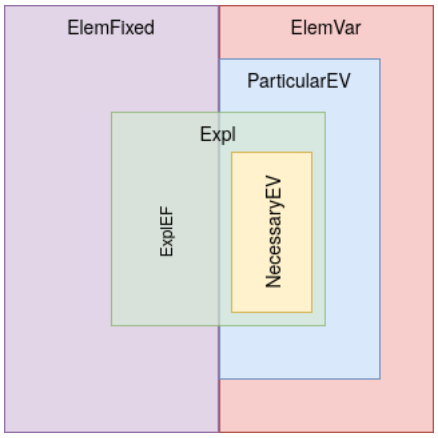

Figure 5.1: Inclusion relations of predicates that rule the presence or absence of elements in an explanation of the first group in Axioms (5.2)

We begin with the formulas that describe the first group of explanations, that is to say the explanations for the Coherence and *Rein1* principles.

First group of Explanations

$$\forall x \in ElemFixed \ \ (Expl(x) \leftrightarrow ExplEF(x)) \tag{5.2a}$$

$$\forall x \in ElemVar \ \ (Expl(x) \rightarrow ParticularEV(x)) \tag{5.2b}$$

$$\forall x \in ElemVar \ \ ((ParticularEV(x) \wedge NecessaryEV(x)) \rightarrow Expl(x)) \tag{5.2c}$$

$$\begin{aligned} &\big(\exists x \in ElemVar \ \ (ParticularEV(x) \wedge \neg NecessaryEV(x))\big) \rightarrow \\ &\quad \big(\exists x \in ElemVar \ \ (ParticularEV(x) \wedge \neg NecessaryEV(x) \wedge Expl(x))\big) \end{aligned} \tag{5.2d}$$

Looking back at definitions 34 and 38, Formula (5.2a) aims at representing the first condition of Definition 34, and the second condition of Definition 38. Formulas (5.2b) and (5.2c) aim at representing the second condition of Definition 34, and the first condition of Definition 38. Finally, Formula (5.2d) aims at representing the third condition of both Definitions 34 and 38.

As we said, we distinguish between two types of elements in the Argumentation Framework : (i) fixed elements ($ElemFixed$), and (ii) variable elements ($ElemVar$). The distinction between the two is the way they are selected to be kept in the explanation.

For the fixed elements, there is a clear and simple group of them (represented by $ExplEF$), which corresponds directly to those that belong to the explanation, no more, no less.

For the variable elements, there are some that will be mandatory for the explanation, those that may or may not be part of it, and those that will not belong to it. In the case of the first group of explanations, the elements that are mandatory and those that may or may not belong to the explanation are parts of the same group that can be identified ($ParticularEV$). In other terms, among some particular variable elements, some will necessarily belong to the explanation ($NecessaryEV$), while the others (so those that are not $NecessaryEV$) may or may not be part of it.

The diagram of Figure 5.1 summarizes the inclusion relations between the predicates that dictate the presence or absence of elements in the explanation.

The decision process (of keeping elements in the explanation) is thus based on the variable elements. It is an existential criterion. It states that if there exists a variable element of the group that was identified ($ParticularEV$), and is not necessary in the explanation, then it must belong to it, because its presence will change the interpretation of the explanation.

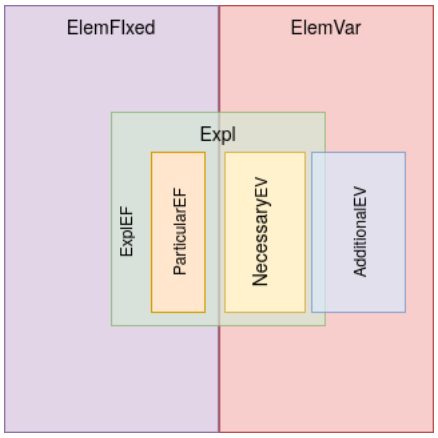

Figure 5.2: Inclusion relations of predicates that rule the presence or absence of elements in an explanation of the second group in Axioms (5.3)

Notation. In the following, considering an AF $\mathcal{A}$ and a set of arguments S, we denote by $\Sigma_1(\mathcal{A}, S)$ the subtheory consisting of $\Sigma(\mathcal{A}, S)$ and Axioms (5.2) together.

We move on to the formulas that describe the second group of explanations, that is to say the explanations for the Defence, *Rein2* and Complement Attacks principles.

Second group of Explanations

$$\forall x \in ElemFixed \ \ (Expl(x) \leftrightarrow ExplEF(x)) \tag{5.3a}$$

$$\forall x \in ElemVar \ \ (NecessaryEV(x) \rightarrow Expl(x)) \tag{5.3b}$$

$$\forall x \in ElemVar \ \ (Expl(x) \rightarrow (NecessaryEV(x) \vee AdditionalEV(x))) \tag{5.3c}$$

$$\begin{aligned} &\forall x \in ParticularEF \ \Big(\big(\exists \alpha, a \ \ (Att(\alpha) \wedge T(\alpha, x) \wedge S(\alpha, a) \wedge Selected(a)\big) \rightarrow \\ &\quad \big(\exists \beta, b \ \ (Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta)\big)\Big) \end{aligned} \tag{5.3d}$$

Looking back at definitions 36, 39 and 41, Formula (5.2a) aims at representing the first condition of each of them. Formulas (5.2b) and (5.2c) aim at representing their second condition. Finally, Formula (5.2d) aims at representing their third condition.

The second group of formulas is similar to the first group to some extent. The fixed elements work in the same way as in the first group of formulas.

However, the variable elements work differently. Instead of having one group of elements, in which some are necessary and the others are not, we consider one group of necessary elements ($NecessaryEV$) and another (potentially different) group of additional elements ($AdditionalEV$). The additional elements are always attacks from the extension to some particular group of arguments taken among the fixed elements ($ParticularEF$). Even if the additional elements are not necessary, some will be made mandatory by Formula (5.3d).

The diagram of Figure 5.2 summarizes the inclusion relations between the predicates that dictate the presence or absence of elements in the explanation.

Remark. As it turns out, in the principles we consider, the fixed elements are always the arguments and the variable elements are always the attacks.

The decision process (of keeping elements in the explanation) is thus based on the arguments. Typically, if some arguments of the group targeted by the additional attacks ($ParticularEF$) are in fact not attacked

by arguments of the extension, it will change the interpretation of the extension. Hence, if such attacks exist, at least one (for each target) is made mandatory by Formula (5.3d).

Notation. In the following, considering an AF $\mathcal{A}$ and a set of arguments S, we denote by $\Sigma_2(\mathcal{A}, S)$ the subtheory consisting of $\Sigma(\mathcal{A}, S)$ and Axioms (5.3) together.

Semantics

We give here the links between the explanations for an extension on an Argumentation Framework and the models of its logical encoding. We begin with a definition whose aim is to characterize an explanation from a model.

Definition 77. Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and $S \subseteq \mathscr{A}$ be a subset of arguments. Let I be an interpretation over $\mathfrak{L}_{Expl}(\mathcal{A})$, we define:

- $\mathscr{A}_I = \{a \in \mathscr{A} \mid I(Expl(a)) = \top\}$
- $\mathscr{R}_I = \{\alpha \in \mathscr{R} \mid I(Expl(\alpha)) = \top\}$
- $s_I : \mathscr{R}_I \mapsto \mathscr{A}_I$ is the function such that $s_I(\alpha) = a$ iff $I(S(\alpha, a)) = \top$[1]
- $t_I : \mathscr{R}_I \mapsto \mathscr{A}_I$ is the function such that $t_I(\alpha) = a$ iff $I(T(\alpha, a)) = \top$[2]

Using a similar approach as in Chapter 4, considering an Argumentation Framework $\mathcal{A}$ and a subset of arguments S in $\mathcal{A}$, we characterize the explanations for S on $\mathcal{A}$ for each Abstract Argumentation principles in terms of the following generic property.

Remark. As in Chapter 4, Property 2 is expressed relatively to some varying sets of formulas, that are intended to represent the specificities of each kind of explanations. Hence, it is a generic property that must be instantiated in order to be applied to each kind of explanations.

Property 2. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and $S \subseteq \mathscr{A}$ be a set of arguments. Let Σ'_{Coh}, Σ'_{Def}, Σ'_{Rein1}, Σ'_{Rein2}, Σ'_{CA} be sets of formulas over $\mathfrak{L}_{Expl}(\mathcal{A})$.*

1. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \Sigma'_{Coh}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
2. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \Sigma'_{Def}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
3. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \Sigma'_{Rein1}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
4. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \Sigma'_{Rein2}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
5. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \Sigma'_{CA}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*

The point is now to specify Σ'_{Coh}, Σ'_{Def}, Σ'_{Rein1}, Σ'_{Rein2}, Σ'_{CA} so that Properties 2.1 to 2.5 indeed hold.

[1] So if $I(S(\alpha, a)) = \bot$ then α has no image by the function s_I.
[2] So if $I(T(\alpha, a)) = \bot$ then α has no image by the function t_I.

5.3.2 Theory for the Coherence Principle

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to explanations for Coherence. For the Coherence principle, the parameters of the generic theory are axiomatized as follows.

Parameters for Coherence

$$\forall x \ (ElemFixed(x) \leftrightarrow Arg(x)) \tag{5.4a}$$

$$\forall x \ (ElemVar(x) \leftrightarrow Att(x)) \tag{5.4b}$$

$$\forall x \ (ExplEF(x) \leftrightarrow Selected(x)) \tag{5.4c}$$

$$\forall x \ \Big(ParticularEV(x) \leftrightarrow \exists a, b\big(S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge Selected(b)\big)\Big) \tag{5.4d}$$

$$\forall x \ (NecessaryEV(x) \leftrightarrow \bot) \tag{5.4e}$$

Formulas (5.4a) and (5.4b) specify that the fixed elements are the arguments and that the variable elements are the attacks respectively. Formula (5.4c) indicates that the fixed elements (so the arguments) that are kept in the explanation are exactly those that belong to the extension. Formula (5.4d) states that the variable elements (so the attacks) that are of interest for this explanation are those between two arguments of the extension. Finally, Formula (5.4e) tells us that among these attacks of interest, none is strictly necessary for the explanation.

Remark. There is no axiomatization of parameters $AdditionalEV$ and $ParticularEF$ because they are not relevant for the first group of explanations (i.e. they do not appear in Axioms (5.2)).

Note that (5.4) can be used to rewrite Formulas (5.2), as formulas $((5.2a)_{\text{Coh}})$, $((5.2b)_{\text{Coh}})$, $((5.2c)_{\text{Coh}})$ and $((5.2d)_{\text{Coh}})$ below.

Explanation for Coherence

$$\forall x \in Arg \ (Expl(x) \leftrightarrow Selected(x)) \tag{$(5.2a)_{\text{Coh}}$}$$

$$\forall x \in Att \ \big(Expl(x) \rightarrow \big[\exists a, b(S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge Selected(b))\big]\big) \tag{$(5.2b)_{\text{Coh}}$}$$

$$\forall x \in Att \ (\bot \rightarrow Expl(x))^3 \tag{$(5.2c)_{\text{Coh}}$}$$

$$\big(\exists x \in Att \ (\exists a, b(S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge Selected(b)))\big) \rightarrow \big(\exists x \in Att \ (\exists a, b(S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge Selected(b)) \wedge Expl(x))\big) \tag{$(5.2d)_{\text{Coh}}$}$$

5.3.3 Theory for the Defence Principle

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to explanations for Defence. For the Defence principle, the parameters of the generic theory are axiomatized as follows.

[3] Note that Formula $(5.2c)_{\text{Coh}}$ is in fact a tautology and has thus no effect.

Parameters for Defence

$$\forall x \ (ElemFixed(x) \leftrightarrow Arg(x)) \tag{5.5a}$$

$$\forall x \ (ElemVar(x) \leftrightarrow Att(x)) \tag{5.5b}$$

$$\forall x \ \Big(IsAttacker(x) \leftrightarrow \exists \beta, a \ (Att(\beta) \wedge S(\beta, x) \wedge T(\beta, a) \wedge Selected(a))\Big) \tag{5.5c}$$

$$\forall x \ \Big(ExplEF(x) \leftrightarrow (Selected(x) \vee IsAttacker(x))\Big) \tag{5.5d}$$

$$\forall x \ \Big(NecessaryEV(x) \leftrightarrow \exists a, b \ (S(x, b) \wedge T(x, a) \wedge Selected(a) \wedge IsAttacker(b))\Big) \tag{5.5e}$$

$$\forall x \ \Big(AdditionalEV(x) \leftrightarrow \exists a, b \ (S(x, a) \wedge T(x, b) \wedge Selected(a) \wedge IsAttacker(b))\Big) \tag{5.5f}$$

$$\forall x \ \Big(ParticularEF(x) \leftrightarrow IsAttacker(x)\Big) \tag{5.5g}$$

Formulas (5.5a) and (5.5b) specify that the fixed elements are the arguments and that the variable elements are the attacks respectively. Formula (5.5c) defines an attacker as an argument that attacks an argument of the extension. Then, Formula (5.5d) indicates that the fixed elements (so the arguments) that are kept in the explanation are exactly those that belong to the extension and those that attack these arguments. Formula (5.5e) states that the variable elements (so the attacks) that are necessary in the explanation are those that go from an attacker of the extension to an argument of the extension. Formula (5.5f) states that the attacks that may or may not be in the explanation are those that go from an argument of the extension to an attacker of the extension. Finally, Formula (5.5g) tells us that among the arguments, those that are of interest for this explanation are the attackers of the extension.

Remark. The predicate $IsAttacker$ is here a mere alias that is introduced for the sole purpose of improving readability. It is not necessary in the theory.

Remark. There is no axiomatization of the parameter $ParticularEV$ because it is not relevant for the second group of explanations (i.e. it does not appear in Axioms (5.3)).

Note that (5.5) can be used to rewrite Formulas (5.3), as Formulas $((5.3a)_{\text{Def}})$, $((5.3b)_{\text{Def}})$, $((5.3c)_{\text{Def}})$ and $((5.3d)_{\text{Def}})$ below (keeping Formula (5.5c) for the definition of the predicate $IsAttacker$).

Explanation for Defence

$$\forall x \in Arg \ (Expl(x) \leftrightarrow (Selected(x) \vee IsAttacker(x))) \tag{$(5.3a)_{\text{Def}}$}$$

$$\forall x \in Att \ \Big((\exists a, b \ (S(x, b) \wedge T(x, a) \wedge Selected(a) \wedge IsAttacker(b))) \rightarrow Expl(x)\Big) \tag{$(5.3b)_{\text{Def}}$}$$

$$\forall x \in Att \ \Big(Expl(x) \rightarrow \big(\exists a, b \ (S(x, b) \wedge T(x, a) \wedge Selected(a) \wedge IsAttacker(b)) \vee \exists a, b \ (S(x, a) \wedge T(x, b) \wedge Selected(a) \wedge IsAttacker(b))\big)\Big) \tag{$(5.3c)_{\text{Def}}$}$$

$$\forall x \in IsAttacker \ \Big(\big(\exists \alpha, a \ (Att(\alpha) \wedge T(\alpha, x) \wedge S(\alpha, a) \wedge Selected(a)\big) \rightarrow \big(\exists \beta, b \ (Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta))\big)\Big) \tag{$(5.3d)_{\text{Def}}$}$$

$$\forall x \ \Big(IsAttacker(x) \leftrightarrow \exists \beta, a \ (Att(\beta) \wedge S(\beta, x) \wedge T(\beta, a) \wedge Selected(a))\Big) \tag{5.5c}$$

5.3.4 Theory for the *Rein1* Principle

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to explanations for *Rein1*. For the *Rein1* principle, the parameters of the generic theory are axiomatized as follows.

Parameters for *Rein1*

$$\forall x \ (ElemFixed(x) \leftrightarrow Att(x)) \tag{5.6a}$$
$$\forall x \ (ElemVar(x) \leftrightarrow Arg(x)) \tag{5.6b}$$
$$\forall x \ (ExplEF(x) \leftrightarrow \bot) \tag{5.6c}$$
$$\forall x \ \Big(ParticularEV(x) \leftrightarrow \forall \alpha \in Att \ \big(\neg T(\alpha, x)\big)\Big) \tag{5.6d}$$
$$\forall x \ (NecessaryEV(x) \leftrightarrow Selected(x)) \tag{5.6e}$$

Formulas (5.6a) and (5.6b) specify that the fixed elements are the attacks and that the variable elements are the arguments respectively. Formula (5.6c) indicates that none of the fixed elements (so the attacks) are kept in the explanation. Formula (5.6d) states that the variable elements (so the arguments) that are of interest for this explanation are those that are not attacked. Finally, Formula (5.6e) tells us that among these arguments of interest, those that are strictly necessary for the explanation are those that belong to the extension.

Remark. As in the case of the Coherence principle, parameters $AdditionalEV$ and $ParticularEF$ are not axiomatized because they are not relevant for the first group of explanations.

Note that (5.6) can be used to rewrite Formulas (5.2), as Formulas $((5.2a)_{\text{Rein1}})$, $((5.2b)_{\text{Rein1}})$, $((5.2c)_{\text{Rein1}})$ and $((5.2d)_{\text{Rein1}})$ below.

Explanation for *Rein1*

$$\forall x \in Att \ (Expl(x) \leftrightarrow \bot) \tag{$(5.2a)_{\text{Rein1}}$}$$
$$\forall x \in Arg \ \big(Expl(x) \rightarrow \big[\forall \alpha \in Att \ (\neg T(\alpha, x))\big]\big) \tag{$(5.2b)_{\text{Rein1}}$}$$
$$\forall x \in Arg \ \big((\forall \alpha \in Att \ (\neg T(\alpha, x)) \wedge Selected(x)) \rightarrow Expl(x)\big) \tag{$(5.2c)_{\text{Rein1}}$}$$
$$\begin{aligned}&\big(\exists x \in Arg \ (\forall \alpha \in Att \ (\neg T(\alpha, x)) \wedge \neg Selected(x))\big) \rightarrow \\ &\quad \big(\exists x \in Arg \ (\forall \alpha \in Att \ (\neg T(\alpha, x)) \wedge \neg Selected(x) \wedge Expl(x))\big)\end{aligned} \tag{$(5.2d)_{\text{Rein1}}$}$$

5.3.5 Theory for the *Rein2* Principle

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to explanations for *Rein2*. For the *Rein2* principle, the parameters of the generic theory are axiomatized as follows.

Parameters for *Rein2*

$$\forall x \ (ElemFixed(x) \leftrightarrow Arg(x)) \tag{5.7a}$$

$$\forall x \ (ElemVar(x) \leftrightarrow Att(x)) \tag{5.7b}$$

$$\forall x \ \Big(IsDefended(x) \leftrightarrow \exists \alpha, \beta, a, b \ (Att(\alpha) \wedge Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge T(\alpha, b) \wedge S(\alpha, a) \wedge Selected(a))\Big) \tag{5.7c}$$

$$\forall x \ \Big(IsAttackerOfDefended(x) \leftrightarrow \exists \gamma, c \ (Att(\gamma) \wedge T(\gamma, c) \wedge S(\gamma, x) \wedge IsDefended(c))\Big) \tag{5.7d}$$

$$\forall x \ \Big(ExplEF(x) \leftrightarrow (Selected(x) \vee IsDefended(x) \vee IsAttackerOfDefended(x))\Big) \tag{5.7e}$$

$$\forall x \ \Big(NecessaryEV(x) \leftrightarrow \exists b, c \ (S(x, b) \wedge T(x, c) \wedge IsAttackerOfDefended(b) \wedge IsDefended(c))\Big) \tag{5.7f}$$

$$\forall x \ \Big(AdditionalEV(x) \leftrightarrow \exists a, b \ (S(x, a) \wedge T(x, b) \wedge Selected(a) \wedge IsAttackerOfDefended(b))\Big) \tag{5.7g}$$

$$\forall x \ \Big(ParticularEF(x) \leftrightarrow IsAttackerOfDefended(x)\Big) \tag{5.7h}$$

Formulas (5.7a) and (5.7b) specify that the fixed elements are the arguments and that the variable elements are the attacks respectively. Formula (5.7c) defines a defended argument as an argument with one of its attackers being attacked by an argument of the extension, while Formula (5.7d) defines an attacker of a defended argument as an argument that attacks a defended argument. Then, Formula (5.7e) indicates that the fixed elements (so the arguments) that are kept in the explanation are exactly those that belong to the extension, those that are defended, and those that attack a defended argument. Formula (5.7f) states that the variable elements (so the attacks) that are necessary in the explanation are those that go from an attacker of a defended argument to a defended argument. Formula (5.7g) states that the attacks that may or may not be in the explanation are those that go from an argument of the extension to an attacker of a defended argument. Finally, Formula (5.7h) tells us that among the arguments, those that are of interest for this explanation are the attackers of a defended argument.

Remark. The predicates $IsDefended$ and $IsAttackerOfDefended$ are here mere aliases that are introduced for the sole purpose of improving readability. They are not necessary in the theory.

Remark. As in the case of the Defence principle, the parameter $ParticularEV$ is not axiomatized because it is not relevant for the second group of explanations.

Note that (5.7) can be used to rewrite Formulas (5.3), as Formulas $((5.3a)_{\text{Rein2}})$, $((5.3b)_{\text{Rein2}})$, $((5.3c)_{\text{Rein2}})$ and $((5.3d)_{\text{Rein2}})$ below (keeping formulas (5.7c) and (5.7d) for the definition of the predicates $IsDefended$ and $IsAttackerOfDefended$).

Explanation for *Rein2*

$$\forall x \in Arg \ \ (Expl(x) \leftrightarrow (Selected(x) \vee IsDefended(x) \vee IsAttackerOfDefended(x))) \quad ((5.3a)_{\text{Rein2}})$$

$$\forall x \in Att \ \ \Big(\big(\exists b, c \ \ (S(x,b) \wedge T(x,c) \wedge IsAttackerOfDefended(b) \wedge IsDefended(c))\big) \rightarrow Expl(x)\Big) \quad ((5.3b)_{\text{Rein2}})$$

$$\forall x \in Att \ \ \Big(Expl(x) \rightarrow \big(\exists b, c \ \ (S(x,b) \wedge T(x,c) \wedge IsAttackerOfDefended(b) \wedge IsDefended(c)) \vee \exists a, b \ \ (S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge IsAttackerOfDefended(b))\big)\Big) \quad ((5.3c)_{\text{Rein2}})$$

$$\forall x \in IsAttackerOfDefended \ \ \Big(\big(\exists \alpha, a \ \ (Att(\alpha) \wedge T(\alpha, x) \wedge S(\alpha, a) \wedge Selected(a)\big) \rightarrow \big(\exists \beta, b \ \ (Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta))\big)\Big) \quad ((5.3d)_{\text{Rein2}})$$

$$\forall x \ \ \Big(IsDefended(x) \leftrightarrow \exists \alpha, \beta, a, b \ \ (Att(\alpha) \wedge Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge T(\alpha, b) \wedge S(\alpha, a) \wedge Selected(a))\Big) \quad (5.7c)$$

$$\forall x \ \ \Big(IsAttackerOfDefended(x) \leftrightarrow \exists \gamma, c \ \ (Att(\gamma) \wedge T(\gamma, c) \wedge S(\gamma, x) \wedge IsDefended(c))\Big) \quad (5.7d)$$

5.3.6 Theory for the Complement Attack Principle

In this section, we present how our generic theory can be parameterized to obtain a theory that corresponds to explanations for Complement Attack. For the Complement Attack principle, the parameters of the generic theory are axiomatized as follows.

Parameters for Complement Attack

$$\forall x \ \ (ElemFixed(x) \leftrightarrow Arg(x)) \quad (5.8a)$$

$$\forall x \ \ (ElemVar(x) \leftrightarrow Att(x)) \quad (5.8b)$$

$$\forall x \ \ \Big(ExplEF(x) \leftrightarrow \top\Big) \quad (5.8c)$$

$$\forall x \ \ \Big(NecessaryEV(x) \leftrightarrow \bot\Big) \quad (5.8d)$$

$$\forall x \ \ \Big(AdditionalEV(x) \leftrightarrow \exists a, b \ \ (S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge \neg Selected(b))\Big) \quad (5.8e)$$

$$\forall x \ \ \Big(ParticularEF(x) \leftrightarrow (Arg(x) \wedge \neg Selected(x))\Big) \quad (5.8f)$$

Formulas (5.8a) and (5.8b) specify that the fixed elements are the arguments and that the variable elements are the attacks respectively. Formula (5.8c) indicates that all the fixed elements (so the arguments) are kept in the explanation. Formula (5.8d) states that no variable element (so the attacks) is necessary in the explanation. Formula (5.8e) states that the attacks that may or may not be in the explanation are those that go from an argument of the extension to an argument that is not in the extension. Finally, Formula (5.8f) tells us that among the arguments, those that are of interest for this explanation are the arguments that do not belong to the extension.

Remark. As in the case of the Defence principle, the parameter $ParticularEV$ is not axiomatized because it is not relevant for the second group of explanations.

Note that (5.8) can be used to rewrite Formulas (5.3), as Formulas ($(5.3a)_{CA}$), ($(5.3b)_{CA}$), ($(5.3c)_{CA}$) and ($(5.3d)_{CA}$) below.

Explanation for Complement Attack

$$\forall x \in Arg \ \ (Expl(x) \leftrightarrow \top) \tag{$(5.3a)_{CA}$}$$

$$\forall x \in Att \ \ (\bot \rightarrow Expl(x))^4 \tag{$(5.3b)_{CA}$}$$

$$\forall x \in Att \ \ (Expl(x) \rightarrow \exists a, b \ \ (S(x,a) \wedge T(x,b) \wedge Selected(a) \wedge \neg Selected(b))) \tag{$(5.3c)_{CA}$}$$

$$\forall x \in (Arg \wedge \neg Selected) \ \ \Big(\big(\exists \alpha, a \ \ (Att(\alpha) \wedge T(\alpha, x) \wedge S(\alpha, a) \wedge Selected(a)\big) \rightarrow \big(\exists \beta, b \ \ (Att(\beta) \wedge T(\beta, x) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta)\big)\Big) \tag{$(5.3d)_{CA}$}$$

5.4 Results

The following theorem is a correct instantiating of Property 2 using the formulas given above.

Theorem 12. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
2. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.5), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
3. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.6), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
4. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.7), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
5. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.8), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*

Unfortunately, and due to limited time constraints, we were not able to provide additional results that go in the direction of the two objectives we fixed at the beginning of the chapter beyond Theorem 12. We consider Theorem 12 to be a primordial result as it demonstrates the correctness of the logical encoding presented in this chapter. However, we also believe that results which go along the way of the two objectives mentioned at the beginning of the chapter should be sought for, as they would fully integrate and complete the entire done in the present work as a whole. Thus, due to our time constraints, the pursue of such results is, for now, left to future considerations.

5.5 Recap example

As for previous chapters, we illustrate the work done in this chapter through an example. This example is an extension of the example presented in Section 3.6.2. Recall that the example was about a political debate held between two candidates. The objective is to identify a satisfying conclusion to the debate among the propositions of the two candidates. So, a computer program has been used to model the debate with an Argumentation Framework (Figure 3.42), and it has been decided that a satisfying conclusion would correspond to a stable extension of that Argumentation Framework.

[4] Note that Formula $(5.3b)_{CA}$ is in fact a tautology and has thus no effect.

In Section 3.6.2, we saw that $\{b, c, g, i, l, p\}$ was a suitable extension, and we showed how the explanations defined in Chapter 3 would be used to convince users of its adequacy. Although it is perfectly possible to get these explanations via Graph Theoretic operations (induced/spanning subgraph operators), we will show here how they can be computed using our logical encoding.

To keep on using previous material, we will show in this section how our logical encoding can be used to compute the explanations of Figures 3.44 and 3.45. They respectively depict an explanation for Coherence and for Complement Attack for $\{b, c, g, i, l, p\}$. The way such explanations can be computed is formalised in Property 2, and more specifically Properties 2.1 and 2.5. In particular, we will in fact use Theorem 12 and more specifically its points 1 and 5 in the case of explanations for Coherence and Complement Attack.

Points 1 and 5 of Theorem 12 tell us that in order to obtain the explanations of Figures 3.44 and 3.45, we must compute a model of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$ (resp.$\Sigma_2(\mathcal{A}, S) \cup \{(5.8), (4.11), (4.12)\}$) in which the predicate $Expl$ is true for the elements present on each Figure, and only them. Recall that $\Sigma_1(\mathcal{A}, S)$ (resp. $\Sigma_2(\mathcal{A}, S)$) is made of $\Sigma(\mathcal{A}, S)$ (itself made of $\Sigma(\mathcal{A})$ and Axiom (5.1)) and Axiom (5.2) (resp. Axiom (5.3)).

Now that the general idea has been discussed, we can see more in detail how our models are computed. Firstly, our models must satisfy $\Sigma(\mathcal{A})$, which is made of Axioms (4.1), (4.2), and (4.3). The same consequences as in Section (4.6.2) will thus follow: we will have the value for the predicates Arg, Att and Sup for the elements of the domain, as well as some values for the predicates S and T. In addition to Axioms (4.1), (4.2), and (4.3), our models must also satisfy Axiom (5.1). The role of this axiom is to encode the extension at hand which, in the case of the problem of computing an explanation, is an input and not an output as in Chapter 4. As such, our models will have given values for the predicate $Selected$ for the elements of the domain. All of this is grouped in Tables 5.2 and 5.3 for the unary predicates. In the case of the binary predicates S and T, Tables 4.8 to 4.11 can be reused (since we are using the same Argumentation Framework). For the same reason, Tables 5.2 and 5.3 are similar to Tables 4.6 and 4.7 in the sense that they share the same values for the predicates Arg, Att and Sup.

Tables 5.2 and 5.3 display the basis on which all the models corresponding to an explanation will be computed. Starting from there, if we firstly consider the explanation of Figure 3.44, our model still needs to satisfy formulas (5.2) and (5.4). We see that formulas (5.4) give us values that we can immediately fill for the predicates $ElemFixed$, $ElemVar$, $ExplEF$ and $NecessaryEV$. As such, we can already deduce from formulas (5.2) that all the arguments will have the predicate $Expl$ true if $Selected$ is true, and false otherwise. Additionally, the formula (5.2c) is trivially satisfied. Regarding the predicate $ParticularEV$, we see from formulas (5.4) that it is true for an element if and only if the element has a source and a target such that $Selected$ is true for both of them. In our setting, using Tables 4.8 to 4.11, we see that this is the case for none of the attacks (recall that by Axioms (4.11) and (4.12), the lines in which there is a "✓" must otherwise be filled with "X", even if it is not displayed for readability reasons). So, $ParticularEV$ is going to be false for all our attacks. However, using formula (5.2b), we have that $Expl$ can be true for an attack only if $ParticularEV$ is true as well, so $Expl$ will be false for all attacks. For the same reason, formula (5.2d) is trivially satisfied. This leads us to the model described by Tables 5.4 and 5.5, which corresponds with the explanation of Figure 3.44.

If we move on to the explanation of Figure 3.45, we can start from the model of Tables 5.2 and 5.3. The model then also needs to satisfy formulas (5.3) and (5.8). As before, formulas (5.8) immediately give use values for some predicates, namely $ElemFixed$, $ElemVar$, $ExplEF$, $NecessaryEV$ and $ParticularEF$. From there, formulas (5.3) tell us that the predicate $Expl$ will have the value true for all the arguments. Notice as well that formula (5.3b) is going to be trivially satisfied. Using formula (5.3d), we will look at the arguments that are not in the extension, like for instance a. We see on Tables 4.8 to 4.11 that there exists only one attack (τ) that targets a with its source in the extension (c). So, to satisfy formula (5.3d), $Expl$ must be true for τ. However, we must check that it is coherent with formula (5.3c). This would require that the source of τ is in the extension while its target is not, which is indeed the case. If we look at another argument that is not in the extension, like e, we see on Tables 4.8 to 4.11 that there exist two attacks (ψ and ω) that target e with their source in the extension (c and g). To satisfy formula (5.3d), $Expl$ must then be

	Arg	Att	Sup	$PrimaFacie$	$Selected$	$Expl$	$ElemFixed$	$ElemVar$
a	✓	X	X	✓	X			
b	✓	X	X	✓	✓			
c	✓	X	X	✓	✓			
d	✓	X	X	✓	X			
e	✓	X	X	✓	X			
f	✓	X	X	✓	X			
g	✓	X	X	✓	✓			
h	✓	X	X	✓	X			
i	✓	X	X	✓	✓			
j	✓	X	X	✓	X			
k	✓	X	X	✓	X			
l	✓	X	X	✓	✓			
m	✓	X	X	✓	X			
n	✓	X	X	✓	X			
o	✓	X	X	✓	X			
p	✓	X	X	✓	✓			
α	X	✓	X	✓	X			
β	X	✓	X	✓	X			
γ	X	✓	X	✓	X			
δ	X	✓	X	✓	X			
ε	X	✓	X	✓	X			
η	X	✓	X	✓	X			
θ	X	✓	X	✓	X			
κ	X	✓	X	✓	X			
λ	X	✓	X	✓	X			
μ	X	✓	X	✓	X			
ν	X	✓	X	✓	X			
ξ	X	✓	X	✓	X			
π	X	✓	X	✓	X			
ρ	X	✓	X	✓	X			
σ	X	✓	X	✓	X			
τ	X	✓	X	✓	X			
υ	X	✓	X	✓	X			
ϕ	X	✓	X	✓	X			
χ	X	✓	X	✓	X			
ψ	X	✓	X	✓	X			
ω	X	✓	X	✓	X			
α'	X	✓	X	✓	X			
β'	X	✓	X	✓	X			
γ'	X	✓	X	✓	X			
δ'	X	✓	X	✓	X			
ε'	X	✓	X	✓	X			
η'	X	✓	X	✓	X			
θ'	X	✓	X	✓	X			

Table 5.2: Starting point of a model of $\Sigma_1(\mathcal{A}, S)$ or $\Sigma_2(\mathcal{A}, S)$ for the Argumentation Framework of Figure 4.3 using formulas (4.1), (4.2), (5.1)
(✓: True, X: False)

	$ExplEF$	$ParticularEV$	$NecessaryEV$	$AdditionalEV$	$ParticularEF$
a					
b					
c					
d					
e					
f					
g					
h					
i					
j					
k					
l					
m					
n					
o					
p					
α					
β					
γ					
δ					
ε					
η					
θ					
κ					
λ					
μ					
ν					
ξ					
π					
ρ					
σ					
τ					
υ					
ϕ					
χ					
ψ					
ω					
α'					
β'					
γ'					
δ'					
ε'					
η'					
θ'					

Table 5.3: Starting point of a model of $\Sigma_1(\mathcal{A}, S)$ or $\Sigma_2(\mathcal{A}, S)$ for the Argumentation Framework of Figure 4.3 using formulas (4.1), (4.2), (5.1) (Continued)

	Arg	Att	Sup	$PrimaFacie$	$Selected$	$Expl$	$ElemFixed$	$ElemVar$
a	✓	X	X	✓	X	X	✓	X
b	✓	X	X	✓	✓	✓	✓	X
c	✓	X	X	✓	✓	✓	✓	X
d	✓	X	X	✓	X	X	✓	X
e	✓	X	X	✓	X	X	✓	X
f	✓	X	X	✓	X	X	✓	X
g	✓	X	X	✓	✓	✓	✓	X
h	✓	X	X	✓	X	X	✓	X
i	✓	X	X	✓	✓	✓	✓	X
j	✓	X	X	✓	X	X	✓	X
k	✓	X	X	✓	X	X	✓	X
l	✓	X	X	✓	✓	✓	✓	X
m	✓	X	X	✓	X	X	✓	X
n	✓	X	X	✓	X	X	✓	X
o	✓	X	X	✓	X	X	✓	X
p	✓	X	X	✓	✓	✓	✓	X
α	X	✓	X	✓	X	X	X	✓
β	X	✓	X	✓	X	X	X	✓
γ	X	✓	X	✓	X	X	X	✓
δ	X	✓	X	✓	X	X	X	✓
ε	X	✓	X	✓	X	X	X	✓
η	X	✓	X	✓	X	X	X	✓
θ	X	✓	X	✓	X	X	X	✓
κ	X	✓	X	✓	X	X	X	✓
λ	X	✓	X	✓	X	X	X	✓
μ	X	✓	X	✓	X	X	X	✓
ν	X	✓	X	✓	X	X	X	✓
ξ	X	✓	X	✓	X	X	X	✓
π	X	✓	X	✓	X	X	X	✓
ρ	X	✓	X	✓	X	X	X	✓
σ	X	✓	X	✓	X	X	X	✓
τ	X	✓	X	✓	X	X	X	✓
υ	X	✓	X	✓	X	X	X	✓
ϕ	X	✓	X	✓	X	X	X	✓
χ	X	✓	X	✓	X	X	X	✓
ψ	X	✓	X	✓	X	X	X	✓
ω	X	✓	X	✓	X	X	X	✓
α'	X	✓	X	✓	X	X	X	✓
β'	X	✓	X	✓	X	X	X	✓
γ'	X	✓	X	✓	X	X	X	✓
δ'	X	✓	X	✓	X	X	X	✓
ε'	X	✓	X	✓	X	X	X	✓
η'	X	✓	X	✓	X	X	X	✓
θ'	X	✓	X	✓	X	X	X	✓

Table 5.4: Model of formulas $\Sigma_1(\mathcal{A}, S) \cup \{(5.4)\}$ for the Argumentation Framework of Figure 4.3 (Empty cells could have either a "✓" or an "X" without changing the values already displayed)

	$ExplEF$	$ParticularEV$	$NecessaryEV$	$AdditionalEV$	$ParticularEF$
a	X		X		
b	✓		X		
c	✓		X		
d	X		X		
e	X		X		
f	X		X		
g	✓		X		
h	X		X		
i	✓		X		
j	X		X		
k	X		X		
l	✓		X		
m	X		X		
n	X		X		
o	X		X		
p	✓		X		
α	X	X	X		
β	X	X	X		
γ	X	X	X		
δ	X	X	X		
ε	X	X	X		
η	X	X	X		
θ	X	X	X		
κ	X	X	X		
λ	X	X	X		
μ	X	X	X		
ν	X	X	X		
ξ	X	X	X		
π	X	X	X		
ρ	X	X	X		
σ	X	X	X		
τ	X	X	X		
υ	X	X	X		
ϕ	X	X	X		
χ	X	X	X		
ψ	X	X	X		
ω	X	X	X		
α'	X	X	X		
β'	X	X	X		
γ'	X	X	X		
δ'	X	X	X		
ε'	X	X	X		
η'	X	X	X		
θ'	X	X	X		

Table 5.5: Model of formulas $\Sigma_1(\mathcal{A}, S) \cup \{(5.4)\}$ for the Argumentation Framework of Figure 4.3 (Continued)
(Empty cells could have either a "✓" or an "X" without changing the values already displayed)

true for either ψ or ω. Suppose that it is the case for ψ, then $Expl$ can be either true or false for ω. We will assume that it is true as well. Again, it can be checked that this is coherent with formula (5.3c). Proceeding this way for the rest of the arguments that are not in the extension, we end up with the model of Tables 5.6 and 5.7, which corresponds with the explanation of Figure 3.45.

As for the example of Chapter 4, we bring the attention to the fact that Tables 5.4 to 5.7 are incomplete. Again, the empty cells could be filled with either "✓" or "X" without impacting the model (in the limit of respecting the formulas of course). In addition, we have seen for the explanation of Figure 3.45 that we sometimes have the choice of deciding whether an element is present in the explanation (predicate $Expl$ with value true), like with ψ and ω. We first took ψ and had the choice for ω. We could have decided that $Expl$ was false for ω and still obtain a correct model. The explanation corresponding to this model would then still be a valid explanation for Complement Attack for $\{b, c, g, i, l, p\}$, but not the one depicted on Figure 3.45.

5.6 Future Perspectives

Concerning the work done in this chapter, we of course wish to pursue our efforts in the directions of the objectives we stated and which remain to be done.

The first one is using the generic logical encoding of explanations to study how they could be generalised and extended to the enriched frameworks captured by the logical encoding of Chapter 4. To this end, we believe the closeness between the languages of the encoding of this chapter and of Chapter 4 can certainly be valuable. Actually, we can observe that, in both languages, the predicates the languages disagree on always correspond to formulas using only predicates that the languages agree on. Thus, it should technically be possible to express both encodings using the same language, which would be the common parts of both languages. The formulas used would however then be much more complex and difficult to understand. Thus, it requires additional precise formal study to determine if such an integration of the two languages is indeed possible or not. Ideally, the generalisation of our explanations would only be a matter of playing with the parameters of the encoding presented in this chapter. However, it is not possible to asses now whether it will be needed to modify Formulas (5.2) and (5.3) to capture such a generalisation or not.

The second, and possibly the most important one, is using the generic logical encoding of explanations to seek results relative to the overall evaluation of the explanations' qualities when put in relation with the generic logical encoding of extensions. As we mentioned in our motivation, there exist several notions in formal logic which relate to explanations. Implicants and the abductive reasoning problem are merely examples of such notions. Investigating whether our explanations correspond to some of these notions would give us precious insights on the nature. Such insights could in turn considerably help to categorise our explanations depending on what they are and how they work, so that they can be compared to other notions of explanations.

	Arg	Att	Sup	$PrimaFacie$	$Selected$	$Expl$	$ElemFixed$	$ElemVar$
a	✓	X	X	✓	X	✓	✓	X
b	✓	X	X	✓	✓	✓	✓	X
c	✓	X	X	✓	✓	✓	✓	X
d	✓	X	X	✓	X	✓	✓	X
e	✓	X	X	✓	X	✓	✓	X
f	✓	X	X	✓	X	✓	✓	X
g	✓	X	X	✓	✓	✓	✓	X
h	✓	X	X	✓	X	✓	✓	X
i	✓	X	X	✓	✓	✓	✓	X
j	✓	X	X	✓	X	✓	✓	X
k	✓	X	X	✓	X	✓	✓	X
l	✓	X	X	✓	✓	✓	✓	X
m	✓	X	X	✓	X	✓	✓	X
n	✓	X	X	✓	X	✓	✓	X
o	✓	X	X	✓	X	✓	✓	X
p	✓	X	X	✓	✓	✓	✓	X
α	X	✓	X	✓	X	X	X	✓
β	X	✓	X	✓	X	✓	X	✓
γ	X	✓	X	✓	X	X	X	✓
δ	X	✓	X	✓	X	✓	X	✓
ε	X	✓	X	✓	X	X	X	✓
η	X	✓	X	✓	X	✓	X	✓
θ	X	✓	X	✓	X	✓	X	✓
κ	X	✓	X	✓	X	X	X	✓
λ	X	✓	X	✓	X	X	X	✓
μ	X	✓	X	✓	X	X	X	✓
ν	X	✓	X	✓	X	X	X	✓
ξ	X	✓	X	✓	X	✓	X	✓
π	X	✓	X	✓	X	✓	X	✓
ρ	X	✓	X	✓	X	✓	X	✓
σ	X	✓	X	✓	X	X	X	✓
τ	X	✓	X	✓	X	✓	X	✓
υ	X	✓	X	✓	X	X	X	✓
ϕ	X	✓	X	✓	X	✓	X	✓
χ	X	✓	X	✓	X	X	X	✓
ψ	X	✓	X	✓	X	✓	X	✓
ω	X	✓	X	✓	X	✓	X	✓
α'	X	✓	X	✓	X	X	X	✓
β'	X	✓	X	✓	X	✓	X	✓
γ'	X	✓	X	✓	X	✓	X	✓
δ'	X	✓	X	✓	X	X	X	✓
ε'	X	✓	X	✓	X	X	X	✓
η'	X	✓	X	✓	X	✓	X	✓
θ'	X	✓	X	✓	X	X	X	✓

Table 5.6: Model of formulas $\Sigma_2(\mathcal{A}, S) \cup \{(5.8)\}$ for the Argumentation Framework of Figure 4.3 (Empty cells could have either a "✓" or an "X" without changing the values already displayed)

	$ExplEF$	$ParticularEV$	$NecessaryEV$	$AdditionalEV$	$ParticularEF$
a	✓		X		✓
b	✓		X		X
c	✓		X		X
d	✓		X		✓
e	✓		X		✓
f	✓		X		✓
g	✓		X		X
h	✓		X		✓
i	✓		X		X
j	✓		X		✓
k	✓		X		✓
l	✓		X		X
m	✓		X		✓
n	✓		X		✓
o	✓		X		✓
p	✓		X		X
α	✓		X	X	
β	✓		X	✓	
γ	✓		X	X	
δ	✓		X	✓	
ε	✓		X	X	
η	✓		X	✓	
θ	✓		X	✓	
κ	✓		X	X	
λ	✓		X	X	
μ	✓		X	X	
ν	✓		X	X	
ξ	✓		X	✓	
π	✓		X	✓	
ρ	✓		X	✓	
σ	✓		X	X	
τ	✓		X	✓	
υ	✓		X	X	
ϕ	✓		X	✓	
χ	✓		X	X	
ψ	✓		X	✓	
ω	✓		X	✓	
α'	✓		X	X	
β'	✓		X	✓	
γ'	✓		X	✓	
δ'	✓		X	X	
ε'	✓		X	X	
η'	✓		X	✓	
θ'	✓		X	X	

Table 5.7: Model of formulas $\Sigma_2(\mathcal{A}, S) \cup \{(5.8)\}$ for the Argumentation Framework of Figure 4.3 (Continued)
(Empty cells could have either a "✓" or an "X" without changing the values already displayed)

Chapter 6

Conclusion

This chapter concludes the present work by summarizing our contributions and the different perspectives they open.

Our first main achievement is the definition of visual explanations for Abstract Argumentation. We say "for Abstract Argumention" because indeed, these explanations aim at explaining the basic methods of Abstract Argumentation, which is the selection of arguments via semantics. As we discussed, this is an aspect of Abstract Argumentation that is not the subject of much research from the explainability prism. Most of the work done in this area is focusing on explaining credulous and/or skeptical acceptance, which are mechanisms designed to mitigate one of the drawbacks of some semantics: the very large number of extensions that may exist for a given semantics on a given Argumentation Framework.

These explanations are defined as subgraphs of the initial Argumentation Frameworks, hence their visual nature. Importantly, these explanations are in fact aimed at *answering some questions* (questions that ask for explanations). We have addressed several kinds of questions in the present work. The first kind of questions is probably the most natural one and simply asks for the reasons that makes a set of argument a valid (or not) extension for some semantics on a given Argumentation Framework. The second kind of questions is a bit more complex and is built on the first one. It asks for the reasons which make a set of arguments a valid part (or not) of an arbitrary extension for some semantics on a given Argumentation Framework. It is also possible to include a contrast in the question on a second set of arguments. If the first kind of questions can be seen as challenging a result as a whole, then the second kind of question can be understood as challenging only a part of that result.

It is important to note that our explanations are not defined for all the classical semantics. As it is, only the conflict-free, admissible, complete and stable semantics are studied. Additionally, our explanations are heavily based on the decomposition of semantics into principles. Hence, the general methodology of our approach consists in defining explanations for Abstract Argumentation principles (instead of semantics directly) and then consider that an explanation for a semantics is simply the set of explanations for the principles that compose it.

We have provided a number of results on our explanations that support their use and general desirable behavior. Notably, we proved that the empty explanation, which is an example of undesirable result, only occurs in very specific situations, that may be considered as off the charts from the beginning. Additionally, we have provided insights on the structural organization of explanations, such as the existence of one unique maximal explanation and several minimal explanations, the union of which exactly corresponds to the maximal one, for each Abstract Argumentation principles we studied. However, the most critical results in our opinion are the ones showing that our explanations can be used to answer the questions from which they were defined using only *structural properties* (i.e. properties that can be *seen* on the graph). This is an especially important result because it means that potentially anyone can use these explanations as inspection facilitators for the initial Argumentation Frameworks, *without prior knowledge of Abstract Argumentation mechanisms*. We also provided ways to compute the maximal explanation for each Abstract Argumentation

principle using basic graph theoretic operations, and algorithms to compute any minimal explanation from the maximal one. These algorithms have been proved sound and complete for this purpose, and can be easily adapted to compute any intermediary explanation.

Our second main contribution is the definition of a generic logical encoding to compute Abstract Argumentation semantics that captures both the basic Argumentation Framework and several of its generalisations. This logical encoding is generic in that it is composed of two kinds of formulas: the shared formulas and the instantiating formulas. As their name indicates, the shared formulas are those formulas that are at a somewhat "general" level and are present in all theories derived from the generic logical encoding. Importantly, these shared formulas are written using what we called "parameter" predicates. That is to say, predicates that are not defined at the general level of the shared formulas. The second kind of formulas, the instantiating ones, are those that define and give a meaning to the parameter predicates so that the shared formulas are completely defined. They are aimed at representing the specificities of the Abstract Argumentation Framework that is being encoded. Hence, the association of the shared formulas with a set of formulas that instantiates the parameter predicates yields a complete theory that can be used to compute the extensions of some framework for a classical semantics.

The semantics that can be computed using this logical encoding are all the classical semantics initially defined by Dung. To do so, we take advantage of the decomposition of these semantics into Abstract Argumentation principles. As such, the logical encoding is in fact made of formulas that represent these principles instead of having formulas that directly represent the semantics. This way, the semantics that is being computed is dependent on which formulas of the encoding (i.e. principles) are being used to build the theory. Since the logical encoding covers several different Abstract Argumentation Frameworks, it is important to note that for a given semantics, it is always the same shared formulas that are being used, no matter which specific framework is being handled.

As we said, the generic logical encoding captures Argumentation Frameworks and a certain number of its generalisations. More precisely, the generalisations captured are those that can be obtained using any combinations of the following enrichments: coalitions of arguments, addition of a support relation, and higher-order relations. It is important to note that some restrictions apply: for each enrichment, only a specific interpretation is taken into account for now. Thus, regarding the coalition enrichment, we consider that a relation is effective when all the arguments of its source respect some property, and we consider that the relations cannot target coalition (i.e. targets of relations are always a single argument). Regarding the support relation, we only consider its evidential interpretation, without support cycle when higher-order relations are present. And in the case of higher-order relations, we restrict ourselves to its RAF interpretation. To handle all these enrichments, the logical encoding is given regarding an Abstract Argumentation Framework that possess all of them at once: the framework we called Higher-Order Evidence-Based Argumentation Framework with Coalitions (HO-EBAF-C). Certain restrictions on the general definition of the HO-EBAF-C allow to retrieve the simpler frameworks. Likewise, in the logical encoding, the less enrichments are present, the simpler the instantiating of the parameters is. We have provided several results that give a correspondence between the models of each theory obtained using the different instantiating we studied and the extensions of the corresponding Abstract Argumentation Framework for the considered semantics.

Our third main contribution is the extension of the logical encoding of Chapter 4 to compute our explanations defined in Chapter 3. There were several objectives to achieve by doing so. First of all, it would allow to have both the computation of extensions, and the computation of explanations for these extensions embedded in the same tool. Secondly, since our explanations are only defined for Argumentation Frameworks but the logical encoding to compute extensions is defined for several of their generalisations, expressing the explanations in the same theory will certainly facilitate their definition for enriched frameworks. Finally, having both the explanations and their subject encoded in the same logical theory certainly opens the perspective of studying in depth their relation from the prism of previous works in logical explanations.

Firstly, we consider important to recall that the logical encoding of explanations, while going along the line of the encoding of Chapter 4, is not exactly integrated inside the same theory. Instead, both encoding share a common part (the encoding of the framework) and then have each a specific part (the encoding of Abstract Argumentation principles for one, the encoding of the explanations for the other). As such, it would

be abusive to claim that both encoding are indeed defined in the same logical theory. However, they do have common grounds, and the encoding of the explanations do follow a similar methodology as the encoding of the extensions as it is also generic. That is to say, in the logical encoding of explanations, there are also two kinds of formulas, the shared ones and the instantiating ones, with the same "roles" (but not the same subjects) as in the logical encoding of extensions. This methodology allowed us to uncover the fact that our explanations could in fact be categorized into two different groups according to how they explain their subject: this is what is captured by the shared formulas. The instantiating formulas would the capture the specificities of each kind of explanations inside each different groups.

Similarly to Chapter 4, we have provided a result that establishes a correspondence between the models of the different theories obtained using our instantiating formulas and the different kinds of explanations as defined in Chapter 3. This is a first step towards the generalisation of our explanations to enriched frameworks, as those considered in Chapter 4, and the study of correspondences between our explanations and classical accounts of explanations from a logical perspectives. This objective remains to be fulfilled, but we consider it to be important roads to explore and we firmly intend to continue our work in their direction.

Regarding the different paths left open for future research, our contributions pave the way for a lot of directions. The most obvious ones are on the possible ways of extending our work to capture more possibilities. These include the extension of our explanations from Chapter 3 to cover the missing classical grounded and preferred semantics, as well as to cover generalisations of Abstract Argumentation. The works of Chapters 4 and 5 can certainly help on that last matter. Indeed, the study of the connections between the logical encoding of explanations (without enrichments) and the logical encoding of semantics (with enrichments) should yield a way to adapt the formulas encoding the explanations to take enrichments into account, and so, at the same time, obtain generalised definitions of the explanations. Still, it is not obvious how much the definitions of explanations would be affected by the presence of enrichments, since they directly affect semantics, which are at the core of our explanations. Thus, this line of research could prove to be challenging. Our logical encoding from Chapter 4 can also be extended to capture more diverse interpretations of the generalisations of Abstract Argumentation that are already considered, or even include additional ones. Finally, he logical encoding of Chapter 5 could also be extended to cover generalised aspects of Abstract Argumentation using its proximity with the logical encoding of Chapter 4, which coincides with the first point mentioned just before.

Specifically to Chapter 3, we can also mention the possibility of exploring different visual properties from Graph Theory to exploit in our explanations in order to characterize meaningful argumentative results in terms of visual organization of the Argumentation Framework. We could as well research the definition of explanations as answers to different questions than those addressed in the present work, like questions putting a contrast on other elements of the context, or questions on how the Argumentation Framework was built. There is also the possibility of studying explanations corresponding to a different way of explaining than the one that was presented. We could think for instance of using graph isomorphism to show to a user that the conclusion of a particular area of the graph should be the same as another, isomorphic, of the same graph on which the user agrees. We additionally recall the need to conduct social experiments to confirm or not our intuition that the explanations defined in Chapter 3 can indeed be understood and used by people who are non experts in computer science.

Although we provided examples illustrating how our encodings works and how to use them in Chapters 4 and 5, these still rely to a certain point on some degree of intuition. As such, we could in the future propose additional works presenting in full detail the complete methodology of how to use these encodings.

Lastly, the logical encoding of Chapter 5 should certainly be used to study more deeply the relations our explanations from a logical perspective. In particular, the ways it relates with the logical encoding of Chapter 4 should be of interest. For a given Argumentation Framework, questions such as "Are our explanations implicants of the argumentative process?" or "Are our explanations the results of an abductive reasoning problem?" would certainly provide insights on the nature of the explanations we defined.

Finally, this work on explanations for Abstract Argumentation should be put in a greater perspective of works on explanations. In particular, there exists a vast body of work on the matter in philosophy, psy-

chology, cognitive science, ... which was debated and thus matured for a much longer time than it was in computer science. The survey [Mil19] could prove to be a valuable entry point on the subject for instance. [Mil19] advocates that the findings on the question of explanations in these other fields are relevant for and should be applied in XAI. One of the major results from these fields is the fact that humans tend to project human characteristics on artificial systems. This means in particular that humans would expect a computer program to explain something *in the same way a human would have done*. Hence the relevance of findings on how humans explain. Results on the matter show that humans tend to explain *contrastively*. That is to say, humans tend to explain by generating an alternative scenario from what really happened and *select* the elements of the explanation they deliver in the difference between this alternative scenario and reality. Moreover, humans also tend to be subject to some *biases* when generating such scenarios: they focus on some specific elements. These elements are often based on a duality and may for instance include intentionality/unintentionality, normality/abnormality, necessity/sufficiency, internal/external causes, temporal closeness/distance, or controllability/uncontrollability. A last example of finding is the fact that, to borrow words that are often used in these works: "Explanation is a three-place predicate: *someone* explains *something* to *someone*". As such, part of the explaining process is the transfer of the explanation from one person to another, which is often adapted to the person receiving the explanation.

If computer science is to take inspiration from these works to provide explanations for artificial systems, difficulties will likely include the identification of the proper biases to use when generating counterfactuals, as they may very well vary from one application to another. Also, even though counterfactuals are properly generated, the selection of one of them to serve as the basis of an explanation, and then the selection of elements inside this counterfactual as the proper content of the explanation is not really clear as well. Moreover, the process of adapting a proper explanation to an interlocutor still largely has gray areas. Nonetheless, all the works on explanation from social sciences certainly are closer to be the foundations of a general understanding of explanations than the works from computer science. If a general theory of explanations is to be developed, and we believe it should, then all these different insights can certainly prove to be immensely valuable, and works in computer science should probably start considering taking inspirations from them.

Bibliography

[ABC17] Abdallah Arioua, Patrice Buche, and Madalina Croitoru. Explanatory dialogues with argumentative faculties over inconsistent knowledge bases. *Expert Systems with Applications*, 80:244–262, 2017.

[AGPT20] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. On the semantics of abstract argumentation frameworks: A logic programming approach. *Theory and Practice of Logic Programming*, 20(5):703–718, September 2020.

[AGPT21] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. Defining the semantics of abstract argumentation frameworks through logic programs and partial stable models (extended abstract). In Zhi-Hua Zhou, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4735–4739, Virtual Event / Montreal, Canada, August 2021. IJCAI Organization.

[BB20a] AnneMarie Borg and Floris Bex. Explaining arguments at the dutch national police. In Víctor Rodríguez-Doncel, Monica Palmirani, Michal Araszkiewicz, Pompeu Casanovas, Ugo Pagallo, and Giovanni Sartor, editors, *Proceedings of the International Workshops on AI Approaches to the Complexity of Legal Systems (AICOL), Revised Selected Papers*, volume 13048 of *Lecture Notes in Computer Science*, pages 183–197. Springer, 2020.

[BB20b] AnneMarie Borg and Floris Bex. Necessary and sufficient explanations in abstract argumentation. *Computing Research Repository (CoRR)*, abs/2011.02414, 2020.

[BB21a] AnneMarie Borg and Floris Bex. A basic framework for explanations in argumentation. *Intelligent Systems*, 36(2):25–35, March 2021.

[BB21b] AnneMarie Borg and Floris Bex. Contrastive explanations for argumentation-based conclusions. *Computing Research Repository*, abs/2107.03265, 2021.

[BB21c] AnneMarie Borg and Floris Bex. Necessary and sufficient explanations for argumentation-based conclusions. In Jirina Vejnarová and Nic Wilson, editors, *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, volume 12897 of *Lecture Notes in Computer Science*, pages 45–58, Prague, Czech Republic, September 2021. Springer.

[BCG18] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In Pietro Baroni, Dov M. Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, volume 1, pages 157–234. College Publications, February 2018.

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

[BD04] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In James P. Delgrande and Torsten Schaub, editors, *Proceedings of the International Workshop on Non-Monotonic Reasoning (NMR)*, pages 59–64, Whistler, Canada, June 2004.

[BDDL22] Philippe Besnard, Sylvie Doutre, Théo Duchatelle, and Marie-Christine Lagasquie-Schiex. Question-Based Explainability in Abstract Argumentation. Research Report IRIT/RR–2022–01–FR, IRIT : Institut de Recherche en Informatique de Toulouse, France, 2022.

[BES+18] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract dialectical frameworks. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, volume 1, pages 237–286. College Publications, February 2018.

[BGK+14] Richard Booth, Dov M. Gabbay, Souhila Kaci, Tjitze Rienstra, and Leendert W. N. van der Torre. Abduction and dialogical proof in argumentation and logic programming. In Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, editors, *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 117–122, Prague, Czech Republic, August 2014. IOS Press.

[BGvdTV10] Guido Boella, Dov M. Gabbay, Leendert W. N. van der Torre, and Serena Villata. Support in abstract argumentation. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo R. Simari, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 111–122, Desenzano del Garda, Italy, September 2010. IOS Press.

[BGW05] Howard Barringer, Dov M. Gabbay, and John Woods. *Temporal Dynamics of Support and Attack Networks: From Argumentation to Zoology*, volume 2605 of *Lecture Notes in Computer Science*, pages 59–98. Springer, 2005.

[BM08] Adrian Bondy and Uppaluri S. R. Murty. *Graph Theory*. Graduate Texts in Mathematics. Springer, 2008.

[BU21] Ringo Baumann and Markus Ulbricht. Choices and their consequences - explaining acceptable sets in abstract argumentation frameworks. In Meghyn Bienvenu, Gerhard Lakemeyer, and Esra Erdem, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 110–119, Online event, November 2021. IJCAI Organization.

[CDGV13] Federico Cerutti, Paul E. Dunne, Massimiliano Giacomin, and Mauro Vallati. Computing preferred extensions in abstract argumentation: A sat-based approach. In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Proceedings of the International Workshop on Theory and Applications of Formal Argumentation (TAFA), Revised Selected papers*, volume 8306 of *Lecture Notes in Computer Science*, pages 176–193, Beijing, China, August 2013. Springer.

[CFFL18a] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Argumentation frameworks with recursive attacks and evidence-based supports. In Flavio Ferrarotti and Stefan Woltran, editors, *Proceedings of the International Symposium on Foundations of Information and Knowledge Systems (FoIKS)*, volume 10833 of *Lecture Notes in Computer Science*, pages 150–169, Budapest, Hungary, May 2018. Springer.

[CFFL18b] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Structure-based semantics of argumentation frameworks with higher-order attacks and supports. In Carlos I. et al. Chesñevar, editor, *Argumentation-based Proofs of Endearment. Essays in Honor of Guillermo R. Simari on the Occasion of his 70th Birthday*, volume 37 of *Tributes*, pages 43–72. College Publications, 2018.

[CFFL21] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Valid attacks in argumentation frameworks with recursive attacks. *Annals of Mathematics and Artificial Intelligence (Special Issue: Commonsense 2017)*, 89(1-2):53–101, February 2021.

[CG09] Martin W. A. Caminada and Dov M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2-3):109–145, November 2009.

[CGGS15] Andrea Cohen, Sebastian Gottifredi, Alejandro J. García, and Guillermo R. Simari. An approach to abstract argumentation with recursive attack and support. *Journal of Applied Logic*, 13(4):509–533, December 2015.

[Cho63] Noam Chomsky. Formal properties of grammars. In R. Duncan Luce, Robert R. Bush, and Eugene Galanter, editors, *Handbook of Mathematical Psychology*, volume 2, chapter 12, pages 323–418. John Wiley and Sons, Inc., New York and London, 1963.

[CL05] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In Lluís Godo, editor, *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, volume 3571 of *Lecture Notes in Computer Science*, pages 378–389, Barcelona, Spain, July 2005. Springer.

[CL13] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolarity in argumentation graphs: Towards a better understanding. *International Journal of Approximate Reasoning*, 54(7):876–899, September 2013.

[CL18] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Logical encoding of argumentation frameworks with higher-order attacks. In Lefteri H. Tsoukalas, Éric Grégoire, and Miltiadis Alamaniotis, editors, *Proceedings of the International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 667–674, Volos, Greece, November 2018. IEEE.

[CL20] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Logical encoding of argumentation frameworks with higher-order attacks and evidential supports. *International Journal on Artificial Intelligence Tools*, 29(3-4):2060003:1–2060003:50, June 2020.

[CM63] Noam Chomsky and George A. Miller. Introduction to the formal analysis of natural languages. In R. Duncan Luce, Robert R. Bush, and Eugene Galanter, editors, *Handbook of Mathematical Psychology*, volume 2, chapter 11, pages 269–321. John Wiley and Sons, Inc., New York and London, 1963.

[CNO09] José Luis Carballido, Juan Carlos Nieves, and Mauricio Osorio. Inferring preferred extensions by pstable semantics. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 13(41):38–53, 2009.

[Com21] European Commission. Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts. `https://artificialintelligenceact.eu/the-act/`, 2021.

[ČRA+21] Kristijonas Čyras, Antonio Rago, Emanuele Albini, Pietro Baroni, and Francesca Toni. Argumentative XAI: A survey. In Zhi-Hua Zhou, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4392–4399, Online Event / Montreal, Canada, August 2021. IJCAI Organization.

[DAR16] DARPA. Broad agency announcement : Explainable artificial intelligence (xai). `https://www.darpa.mil/attachments/DARPA-BAA-16-53.pdf`, August 2016.

[DD18] Wolfgang Dvorák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, volume 1, pages 629–688. College Publications, February 2018.

[DDK+23] Yannis Dimopoulos, Wolfgang Dvorák, Matthias König, Anna Rapberger, Markus Ulbricht, and Stefan Woltran. Sets attacking sets in abstract argumentation. In Kai Sauerwald and Matthias Thimm, editors, *Proceedings of the International Workshop on Non-Monotonic Reasoning (NMR)*, volume 3464 of *CEUR Workshop Proceedings*, pages 22–31, Rhodes, Greece, September 2023. CEUR-WS.org.

[DJWW12] Wolfgang Dvořák, Matti Järvisalo, Johannes Peter Wallner, and Stefan Woltran. CEGARTIX: a SAT-based argumentation system. In *Proceedings of the International Workshop on Pragmatics of SAT*, Trento, Italy, June 2012.

[DM16] Sylvie Doutre and Jean-Guy Mailly. Quantifying the difference between argumentation semantics. In Pietro Baroni, Thomas F. Gordon, Tatjana Scheffler, and Manfred Stede, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 287 of *Frontiers in Artificial Intelligence and Applications*, pages 255–262, Potsdam, Germany, September 2016. IOS Press.

[DRW23] Wolfgang Dvorák, Anna Rapberger, and Stefan Woltran. A claim-centric perspective on abstract argumentation semantics: Claim-defeat, principles, and expressiveness. *Artificial Intelligence*, 324:104011, November 2023.

[dSBCL16] Florence Dupin de Saint-Cyr, Pierre Bisquert, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. Argumentation update in YALLA (yet another logic language for argumentation). *International Journal of Approximate Reasoning*, 75:57–92, August 2016.

[Dun95] 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.

[EGW10] Uwe Egly, Sarah Alice Gaggl, and Stefan Woltran. Answer-set programming encodings for argumentation frameworks. *Argument & Computation*, 1(2):147–177, June 2010.

[FB19] Giorgos Flouris and Antonis Bikakis. A comprehensive study of argumentation frameworks with sets of attacking arguments. *International Journal of Approximate Reasoning*, 109:55–86, June 2019.

[FT15a] Xiuyi Fan and Francesca Toni. On computing explanations in argumentation. In Blai Bonet and Sven Koenig, editors, *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) International Conference*, pages 1496–1502, Austin, Texas, USA, January 2015. AAAI Press.

[FT15b] Xiuyi Fan and Francesca Toni. On explanations for non-acceptable arguments. In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Proceedings of the International Workshop on Theory and Applications of Formal Argumentation (TAFA)*, volume 9524 of *Lecture Notes in Computer Science*, pages 112–127, Buenos Aires, Argentina, July 2015. Springer.

[Gab09a] Dov M. Gabbay. Fibring argumentation frames. *Studia Logica*, 93(2-3):231–295, November 2009.

[Gab09b] Dov M. Gabbay. Semantics for higher level attacks in extended argumentation frames Part 1: Overview. *Studia Logica*, 93(2-3):357–381, November 2009.

[GCGS18] Sebastian Gottifredi, Andrea Cohen, Alejandro Javier García, and Guillermo Ricardo Simari. Characterizing acceptability semantics of argumentation frameworks with recursive attack and support relations. *Artificial Intelligence*, 262:336–368, July 2018.

[GG15] Dov M. Gabbay and Michael Gabbay. The attack as strong negation, part I. *Logic Journal of the IGPL*, 23(6):881–941, 2015.

[Har65] Gilbert H. Harman. The inference to the best explanation. *The Philosophical Review*, 74(1):88–95, January 1965.

[Hod13] Richard E. Hodel. *An Introduction to Mathematical Logic*. Dover, 2013.

[KP01] Nikos I. Karacapilidis and Dimitris Papadias. Computer supported argumentation and collaborative decision making: the HERMES system. *Information Systems*, 26(4):259–277, 2001.

[Lag21] Marie-Christine Lagasquie-Schiex. Handling support cycles and collective interactions in the logical encoding of higher-order bipolar argumentation frameworks. In Pietro Baroni, Christoph Benzmüller, and Yì N. Wáng, editors, *Proceedings of the International Conference on Logic and Argumentation (CLAR)*, volume 13040 of *Lecture Notes in Computer Science*, pages 244–265, Hangzhou, China, October 2021. Springer.

[Lag23] Marie-Christine Lagasquie-Schiex. Handling support cycles and collective interactions in the logical encoding of higher-order bipolar argumentation frameworks. *Journal of Logic and Computation*, 33(2):289–318, March 2023.

[LL17] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Proceedings of the Annual Conference on Advances in Neural Information Processing Systems (NIPS)*, volume 30, pages 4765–4774, Long Beach, CA, USA, December 2017.

[LvdT20] Beishui Liao and Leendert van der Torre. Explanation semantics for abstract argumentation. In Henry Prakken, Stefano Bistarelli, Francesco Santini, and Carlo Taticchi, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 271–282, Perugia, Italy, September 2020. IOS Press.

[MC09] Sanjay Modgil and Martin Caminada. Proof theories and algorithms for abstract argumentation frameworks. In Guillermo Ricardo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, chapter 6, pages 105–129. Springer, Boston, MA, 2009.

[Mil19] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[MS88] Johanna D. Moore and William R. Swartout. Explanation in expert systems : A survey. Technical Report ISI/RR-88-228, University of California, Information Science Institute, Marina del Rey, CA, USA, December 1988.

[NJ20] Andreas Niskanen and Matti Järvisalo. Smallest explanations and diagnoses of rejection in abstract argumentation. In Diego Calvanese, Esra Erdem, and Michael Thielscher, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 667–671, Rhodes, Greece, September 2020. IJCAI Organization.

[NP06] Søren Holbech Nielsen and Simon Parsons. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In Nicolas Maudet, Simon Parsons, and Iyad Rahwan, editors, *Proceedings of the International Workshop on Argumentation in*

Multi-Agent Systems (ArgMAS), volume 4766 of *Lecture Notes in Artificial Intelligence*, pages 54–73, Hakodate, Japan, May 2006. Springer.

[NR11] Farid Nouioua and Vincent Risch. Argumentation frameworks with necessities. In Salem Benferhat and John Grant, editors, *Proceedings of the International Conference on Scalable Uncertainty Management (SUM)*, volume 6929 of *Lecture Notes in Artificial Intelligence*, pages 163–176, Dayton, OH, USA, October 2011. Springer.

[OLR10] Nir Oren, Michael Luck, and Chris Reed. Moving between argumentation frameworks. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo R. Simari, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 379–390, Desenzano del Garda, Italy, September 2010. IOS Press.

[ON08] Nir Oren and Timothy J. Norman. Semantics for evidence-based argumentation. In Philippe Besnard, Sylvie Doutre, and Anthony Hunter, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 276–284, Toulouse, France, October 2008. IOS Press.

[ON17] Mauricio Osorio and Juan Carlos Nieves. Range-based argumentation semantics as two-valued models. *Theory and Practice of Logic Programming*, 17(1):75–90, May 2017.

[PO14] Sylwia Polberg and Nir Oren. Revisiting support in abstract argumentation systems. In Simon Parsons, Nir Oren, Chris Reed, and Federico Cerutti, editors, *Proceedings of the International Conference on Computational Models of Argument (COMMA)*, volume 266 of *Frontiers in Artificial Intelligence and Applications*, pages 369–376, Pitlochry, Scotland, September 2014. IOS Press.

[Qui59] Willard V. Quine. On cores and prime implicants of truth functions. *The American Mathematical Monthly*, 66(9):755–760, November 1959.

[Roe97] Neal J. Roese. Counterfactual thinking. *Psychological Bulletin*, 121(1):133–148, 1997.

[Ros58] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[RSG16] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi, editors, *Proceedings of the International Conference on Knowledge Discovery and Data Mining SIGKDD*, volume 22, pages 1135–1144, San Francisco, CA, USA, August 2016. ACM.

[RT21] Teeradaj Racharak and Satoshi Tojo. On explanation of propositional logic-based argumentation system. In Ana Paula Rocha, Luc Steels, and H. Jaap van den Herik, editors, *Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART)*, volume 2, pages 323–332, Online Streaming, February 2021. SCITEPRESS.

[Rud19] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019.

[SA18] Elizabeth I. Sklar and Mohammad Q. Azhar. Explanation through argumentation. In Michita Imai, Tim Norman, Elizabeth I. Sklar, and Takanori Komatsu, editors, *Proceedings of the International Conference on Human-Agent Interaction (HAI)*, pages 277–285, Southampton, United Kingdom, December 2018. ACM.

[SF21] Alexander Steen and David Fuenmayor. A formalisation of abstract argumentation in higher-order logic. *CoRR*, abs/2110.09174, 2021.

[SWW20] Zeynep Gozen Saribatur, Johannes Peter Wallner, and Stefan Woltran. Explaining non-acceptability in abstract argumentation. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang, editors, *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 881–888, Santiago de Compostela, Spain, September 2020. IOS Press.

[UB19] Markus Ulbricht and Ringo Baumann. If nothing is accepted - repairing argumentation frameworks. *Journal of Artificial Intelligence Reseach*, 66:1099–1145, 2019.

[Uni16] European Union. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). `https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679`, April 2016.

[UW21] Markus Ulbricht and Johannes Peter Wallner. Strong explanations in abstract argumentation. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) International Conference*, pages 6496–6504, Online event, February 2021. AAAI Press.

Appendix A

Proofs of Chapter 3

A.1 Conformity Checks and Visual Behavior

A.1.1 Coherence

Theorem. 1 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Coh}(S)$ be an explanation for Coherence for S on $\mathcal{A}$. S is conflict-free if and only if $Expl_{Coh}(S)$ satisfies C_{Coh}.*

Proof. (for Theorem 1) Denote $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ and let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$.

Suppose that S is conflict-free and that there is an attack (a, b) in $(\mathscr{A}', \mathscr{R}')$. By Def. 34, we have that $\mathscr{R}' \subseteq X$, so $a, b \in S$ and that $(a, b) \in \mathscr{R}$. This contradicts Def. 4 on conflict-freeness.

Suppose now that there are no attacks in $(\mathscr{A}', \mathscr{R}')$ and that S is not conflict-free. By Def. 4, there exists $a, b \in S$ such that $(a, b) \in \mathscr{R}$. Thus, $X \neq \varnothing$ and by Condition 3 of Def. 34, $\mathscr{R}' \neq \varnothing$. This contradicts the absence of attacks in $(\mathscr{A}', \mathscr{R}')$. □

A.1.2 Defence

Theorem. 2 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ be a conflict-free set of arguments and $Expl_{Def}(S)$ be an explanation for Defence for S on $\mathcal{A}$. $S \subseteq F_{\mathcal{A}}(S)$ if and only if $Expl_{Def}(S)$ satisfies C_{Def}.*

Proof. (for Theorem 2) Denote $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ and let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$.

Assume that $S \subseteq F_{\mathcal{A}}(S)$. Suppose now that there is a source vertex b in $\mathscr{R}^{-1}(S)$ in $(\mathscr{A}', \mathscr{R}')$. By Def. 31, we have that $\mathscr{R}'^{-1}(b) = \varnothing$, which means there exists no $a \in \mathscr{A}'$ such that $(a, b) \in \mathscr{R}'$. Because S is conflict-free, $S \cap \mathscr{R}^{-1}(S) = \varnothing$. As $b \in \mathscr{R}^{-1}(S)$ and $\mathscr{R}' \subseteq X \cup Y$ (Def. 36), it must be the case that $a \in S$. Hence, there exists no $a \in S$ such that $(a, b) \in \mathscr{R}'$. So following Condition 3 of Def. 36, there exists no $a \in S$ such that $(a, b) \in \mathscr{R}$. As $b \in \mathscr{R}^{-1}(S)$, there exists $c \in S$ such that $(b, c) \in \mathscr{R}$. Hence, we know that there exists $b \in \mathscr{A}$ with $(b, c) \in \mathscr{R}$ for some $c \in S$ and such that there exists no $a \in S$ with $(a, b) \in \mathscr{R}$. This contradicts the assumption that $S \subseteq F_{\mathcal{A}}(S)$.

Assume now that there are no source vertices in $\mathscr{R}^{-1}(S)$ in $(\mathscr{A}', \mathscr{R}')$. Suppose that there is some $c \in S$ such that c is not acceptable wrt S. By Def. 2, this means that there exists $a \in \mathscr{A}$ such that $(a, c) \in \mathscr{R}$ and there is no $b \in S$ with $(b, a) \in \mathscr{R}$. Firstly, notice that by Def. 28, $a \in \mathscr{R}^{-1}(c)$ and so $a \in \mathscr{R}^{-1}(S)$. Secondly, since $c \in S$, $a \in \mathscr{R}^{-1}(S)$ and $(a, c) \in \mathscr{R}$, $(a, c) \in X$ and so, by Def. 36, it holds that $c, a \in \mathscr{A}'$ and $(a, c) \in \mathscr{R}'$. Thus, by assumption, a is not a source vertex in $(\mathscr{A}', \mathscr{R}')$. Subsequently, there exists $b \in \mathscr{A}'$ such that $(b, a) \in \mathscr{R}'$. Since $a \in \mathscr{R}^{-1}(S)$ and S is conflict-free (i.e. $S \cap \mathscr{R}^{-1}(S) = \varnothing$), it holds that $b \in S$. In addition, as $\mathscr{R}' \subseteq \mathscr{R}$, we deduce that $(b, a) \in \mathscr{R}$. Thus, we have $c \in S$ such that c is not acceptable w.r.t. S and for any $a \in \mathscr{A}$ with $(a, c) \in \mathscr{R}$, there is $b \in S$ with $(b, a) \in \mathscr{R}$, a contradiction. □

Proposition. 4 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$ If S is conflict-free, then $Expl_{Def}(S)$ is a bipartite graph with part S.*

Proof. (for Proposition 4) Denote $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ and suppose S is conflict-free. So, $S \cap \mathscr{R}^{-1}(S) = \varnothing$. Since by Def. 36 $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$, S and $\mathscr{R}^{-1}(S)$ then form a partition of $\mathscr{A}'$. According to Def. 27, we must then show that for every $(a, b) \in \mathscr{R}'$, either $a \in S$ and $b \in \mathscr{R}^{-1}(S)$ or $a \in \mathscr{R}^{-1}(S)$ and $b \in S$. This is given by Def. 36. □

A.1.3 Reinstatement

Theorem. 3 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an explanation for Rein2 for S on $\mathcal{A}$. If $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C_{Reins2}, then $F_{\mathcal{A}}(S) \subseteq S$.*

Proof. (for Theorem 3) Denote $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ and $Expl_{Rein2}(S) = (\mathscr{A}'', \mathscr{R}'')$, and let $X_1 = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$, $X_2 = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)),\ c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$.

Assume that $\mathscr{A}' \subseteq S$ and that all vertices in $\mathscr{R}^{+2}(S) \setminus S$ are the endpoint of an arc whose origin is a source vertex in $(\mathscr{A}'', \mathscr{R}'')$. In other words, $\mathscr{A}' \subseteq S$ and for every $x \in \mathscr{R}^{+2}(S) \setminus S$, there exists $y \in \mathscr{A}''$ such that $(y, x) \in \mathscr{R}''$ and $\mathscr{R}''^{-1}(y) = \varnothing$. Consider $a \in F_{\mathcal{A}}(S)$. This means that for every $b \in \mathscr{A}$ such that $(b, a) \in \mathscr{R}$, there exists $c \in S$ with $(c, b) \in \mathscr{R}$. We must show that $a \in S$. Suppose first that a is not attacked in $\mathcal{A}$. That is to say, $\mathscr{R}^{-1}(a) = \varnothing$, and so, $a \in X_1$. By assumption, $\mathscr{A}' \subseteq S$. This means that $(\mathscr{A}' \setminus S) = \varnothing$. In particular, $(\mathscr{A}' \setminus S) \cap X_1 = \varnothing$, so by Condition 3 of Def. 38, we have $(\mathscr{A} \setminus S) \cap X_1 = \varnothing$. Since $a \in X_1$, $X_1 \neq \varnothing$, so we deduce that either $\mathscr{A} \setminus S = \varnothing$ or $X_1 \subseteq S$. In the first case, we conclude $\mathscr{A} \subseteq S$ and thus $a \in S$. In the second case, we have that $X_1 \cap S = X_1$, and so $X_1 \subseteq \mathscr{A}'$. As $\mathscr{A}' \subseteq S$ by assumption, $a \in S$. Suppose now that $\mathscr{R}^{-1}(a) \neq \varnothing$. By Def. 28, we have $a \in \mathscr{R}^{+2}(S)$ and for every $b \in \mathscr{A}$ such that $(b, a) \in \mathscr{R}$, $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. As such, $(b, a) \in X_2$ and $(c, b) \in Y$. So, by Def. 39, we have that $a, b, c \in \mathscr{A}''$ and $(b, a) \in \mathscr{R}''$. In addition, by Condition 3 of Def. 39, as $b \in \mathscr{R}^{+1}(S)$, there exists $(c', b) \in \mathscr{R}''$ with $c' \in S$. Thus, for every $b \in \mathscr{A}''$ such that $(b, a) \in \mathscr{R}''$, $\mathscr{R}''^{-1}(b) \neq \varnothing$. Hence, all $b \in \mathscr{A}''$ such that $(b, a) \in \mathscr{R}''$ are not source vertices. Consequently, by assumption, $a \notin \mathscr{R}^{+2}(S) \setminus S$, but we know that $a \in \mathscr{R}^{+2}(S)$. It follows that $a \in \mathscr{R}^{+2}(S) \cap S$, and thus that $a \in S$. □

Theorem. 4 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an explanation for Rein2 for S on $\mathcal{A}$. If $F_{\mathcal{A}}(S) \subseteq S$, then $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C'_{Reins2}, with C'_{Reins2} being the condition "all the arguments that S defends but are not in S are attacked by a source vertex or an argument of $\mathscr{R}^{+2}(S)$ in $Expl_{Rein2}(S)$".*

Proof. (for Theorem 4) Denote $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ and $Expl_{Rein2}(S) = (\mathscr{A}'', \mathscr{R}'')$, and let $X_1 = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$, $X_2 = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)),\ c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$.

Assume that $F_{\mathcal{A}}(S) \subseteq S$. Suppose now that either $\mathscr{A}' \not\subseteq S$ or there is a vertex a in $\mathscr{R}^{+2}(S) \setminus S$ that is not the endpoint of an arc whose origin is a source vertex or in $\mathscr{R}^{+2}(S)$ in $(\mathscr{A}'', \mathscr{R}'')$. In the first case, by Def. 38 $\mathscr{A}' \subseteq X_1$. So, we have that there exists $x \in \mathscr{A}$ such that $\mathscr{R}^{-1}(x) = \varnothing$ and $x \notin S$. However, by Def. 2, this means that $x \in F_{\mathcal{A}}(S)$ and $x \notin S$, a contradiction. In the second case, we have $a \notin S$ and for every $b \in \mathscr{A}''$ such that $(b, a) \in \mathscr{R}''$, $\mathscr{R}''^{-1}(b) \neq \varnothing$ and $b \notin \mathscr{R}^{+2}(S)$. In other words, $b \notin \mathscr{R}^{+2}(S)$ and there exists $c \in \mathscr{A}''$ with $(c, b) \in \mathscr{R}''$. By Def. 39, $X_2 \subseteq \mathscr{R}'' \subseteq X_2 \cup Y$. So, since $a \in \mathscr{R}^{+2}(S) \setminus S$ and $b \notin \mathscr{R}^{+2}(S)$, we thus know that $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. In addition, also because $b \notin \mathscr{R}^{+2}(S)$, it must be the case that $c \in S$. So, for every $b \in \mathscr{A}''$ such that $(b, a) \in \mathscr{R}''$, there exists $c \in S$ with $(c, b) \in \mathscr{R}''$. By Def. 39 again, we deduce that for every $b \in \mathscr{A}$ such that $(b, a) \in \mathscr{R}$, there exists $c \in S$ with $(c, b) \in \mathscr{R}$. By Def. 2, this means that a is acceptable wrt S and so that $a \in F_{\mathcal{A}}(S)$. Hence, by assumption, $a \in S$, a contradiction. □

Corollary. 2 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ be a set of arguments such that $\mathscr{R}^{+2}(S)$ is conflict-free, $Expl_{Rein1}(S)$ be an explanation for Rein1 for S on $\mathcal{A}$ and $Expl_{Rein2}(S)$ be an ex-*

planation for Rein2 for S on $\mathcal{A}$. $F_{\mathcal{A}}(S) \subseteq S$ if and only if $Expl_{Rein1}(S)$ satisfies C_{Reins1} and $Expl_{Rein2}(S)$ satisfies C_{Reins2}.

Proof. (for Corollary 2) Immediate from The. 3 and 4. □

A.1.4 Complement Attack

Theorem. 5 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{CA}(S)$ be an explanation for Complement Attack for S on $\mathcal{A}$. $A \setminus S \subseteq \mathscr{R}^{+1}(S)$ if and only if $Expl_{CA}(S)$ satisfies C_{CA}.*

Proof. (for Theorem 5) Denote $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ and let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$.

Assume that $\mathscr{A} \setminus S \subseteq \mathscr{R}^{+1}(S)$. Suppose now that there is an isolated vertex a in $\mathscr{A}' \setminus S$. By Def. 41, we know that $\mathscr{A}' = \mathscr{A}$. As such, there exists $a \in \mathscr{A} \setminus S$ such that a is isolated in $(\mathscr{A}', \mathscr{R}')$. By Def. 33, this means in particular that $\mathscr{R}'^{-1}(a) = \varnothing$ and thus that there is no $b \in \mathscr{A}'$ with $(b, a) \in \mathscr{R}'$. Again, in particular, we have that there is no $b \in S$ with $(b, a) \in \mathscr{R}'$. However, by Condition 3 of Def. 41, we deduce that there is no $b \in S$ with $(b, a) \in \mathscr{R}$. Since $a \in \mathscr{A} \setminus S$, this contradicts the assumption that $\mathscr{A} \setminus S \subseteq \mathscr{R}^{+1}(S)$.

Suppose now that there are no isolated vertices in $\mathscr{A}' \setminus S$ in $(\mathscr{A}', \mathscr{R}')$ and that $\mathscr{A} \setminus S \not\subseteq \mathscr{R}^{+1}(S)$. From the first assumption, by Def. 41, we have that there are no isolated vertices in $\mathscr{A} \setminus S$ in $(\mathscr{A}', \mathscr{R}')$. In particular, by Def. 33, we know that there is no $a \in \mathscr{A} \setminus S$ such that $\mathscr{R}'^{-1}(a) = \varnothing$, or equivalently, for every $a \in \mathscr{A} \setminus S$, there exists $b \in \mathscr{A}$ such that $(b, a) \in \mathscr{R}'$. By Def. 41, we have that $\mathscr{R}' \subseteq X$, thus we deduce that for every $a \in \mathscr{A} \setminus S$, there exists $b \in S$ such that $(b, a) \in \mathscr{R}'$. From the second assumption, we have that there exists some $c \in \mathscr{A} \setminus S$ such that there is no $b \in S$ with $(b, c) \in \mathscr{R}$. By Def. 41 (Conditions 1 and 2), we deduce that there exists some $c \in \mathscr{A} \setminus S$ such that there is no $b \in S$ with $(b, c) \in \mathscr{R}'$, a contradiction of the first assumption. □

Proposition. 5 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$. $Expl_{CA}(S)$ is a bipartite graph with part S and every argument of S is a source vertex in $Expl_{CA}(S)$.*

Proof. (for Proposition 5) Denote $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ and let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. By Def. 41, we have that $\mathscr{A}' = \mathscr{A}$ and $\mathscr{R}' \subseteq X$. An obvious partition of $\mathscr{A}$ based on S is of course S and $\mathscr{A} \setminus S$. As $\mathscr{R}' \subseteq X$, by Def. 27, $(\mathscr{A}', \mathscr{R}')$ is a bipartite graph. In addition, since there is no $(b, a) \in \mathscr{R}'$ such that $b \in \mathscr{A} \setminus S$ and $a \in S$, it holds that for every $a \in S$, $\mathscr{R}'^{-1}(a) = \varnothing$. Thus, by Def. 31, every vertex of S is a source vertex in $(\mathscr{A}', \mathscr{R}')$. □

A.2 Properties on the Classes of Explanations

A.2.1 Empty Explanation

Theorem. 6 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework and $S \subseteq \mathscr{A}$. $(\varnothing, \varnothing)$ is an answer to*

1. *Q_{π}^{Ext} for S on $\mathcal{A}$ with $\pi \in \{Coh, Def, Rein2\}$ if and only if $S = \varnothing$.*
2. *Q_{Rein1}^{Ext} for S on $\mathcal{A}$ if and only if $\{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\} = \varnothing$.*
3. *Q_{CA}^{Ext} for S on $\mathcal{A}$ if and only if $\mathcal{A} = (\varnothing, \varnothing)$.*

Proof. (for Theorem 6)

1. Firstly, consider $\pi = Coh$. Suppose that $(\varnothing, \varnothing)$ is an answer to Q_{Coh}^{Ext} for S on $\mathcal{A}$. According to Def. 34, we have that $\mathscr{A}' = \varnothing = S$. So the "only if" part is satisfied. Suppose now that $S = \varnothing$ and consider the empty graph $(\mathscr{A}' = \varnothing, \mathscr{R}' = \varnothing)$. We must prove that it is an answer to Q_{Coh}^{Ext} for S on $\mathcal{A}$. The first condition, $\mathscr{A}' = \varnothing = S$, is respected by supposition. Since $\mathscr{R}' = \varnothing$, the second condition is respected as well. Moreover, since $S = \varnothing$, by definition $X = \varnothing$ and the third condition is trivially satisfied. Thus, the "if" part is satisfied.

Secondly, consider $\pi = \mathit{Def}$. Suppose that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$. By Def. 36, we have that $S \cup \mathscr{R}^{-1}(S) = \varnothing$ and thus $S = \varnothing$. So the "only if" part is satisfied. Suppose now that $S = \varnothing$ and consider the empty graph $(\mathscr{A}' = \varnothing, \mathscr{R}' = \varnothing)$. We must prove that it is an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$. By supposition, $S = \varnothing$, so $\mathscr{R}^{-1}(S) = \varnothing$ and the first condition is respected. Since $\mathscr{R}^{-1}(S) = \varnothing$, the third condition is also respected. Since $S = \varnothing$, by definition $X = \varnothing$ and the second condition is trivially respected. Thus, the "if" part is satisfied.

Lastly, consider $\pi = \mathit{Rein2}$. Suppose that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$. By Def. 39, we have that $S \cup \mathscr{R}^{+1}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) = \varnothing$ and thus $S = \varnothing$. So the "only if" part is satisfied. Suppose now that $S = \varnothing$ and consider the empty graph $(\mathscr{A}' = \varnothing, \mathscr{R}' = \varnothing)$. We must prove that it is an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$. By supposition, $S = \varnothing$, so $\mathscr{R}^{-1}(S) = \varnothing$ and $\mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) = \varnothing$ as well, which means that the first condition is respected. As $\mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) = \varnothing$, the third condition is also respected. Since $S = \varnothing$, by definition $X = \varnothing$ and the second condition is trivially respected. Thus, the "if" part is satisfied.

2. Let $X = \{a \in A \mid \mathscr{R}^{-1}(a) = \varnothing\}$ and assume that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$. Suppose now that $X \neq \varnothing$. By Def. 38, since $\mathscr{A}' = \varnothing$, it must be the case that $S \cap X = \varnothing$. Since $X \subseteq \mathscr{A}$ and $S \subseteq \mathscr{A}$, this means that $(\mathscr{A} \setminus S) \cap X = X$, and so by supposition that $(\mathscr{A} \setminus S) \cap X \neq \varnothing$. So by Condition 3 of Def. 38, $\exists a \in (\mathscr{A} \setminus S) \cap X$ with $a \in \mathscr{A}'$, which contradicts the hypothesis that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$. As such, we deduce $X = \varnothing$. Thus, the "only if" is satisfied.

   Assume now that $X = \varnothing$. Consider the empty graph $(\varnothing, \varnothing)$. We must prove that it is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$. Condition 2 of Def. 38 is obviously respected. Moreover, since $X = \varnothing$, Condition 1 of Def. 38 is also obviously respected. Again, from $X = \varnothing$ we deduce $(\mathscr{A} \setminus S) \cap X = \varnothing$, and thus Condition 2 of Def. 38 is respected as well. Hence, it follows that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$. Thus, the "if" is satisfied.

3. Assume that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$. By Def. 41 we thus know that $\mathscr{A}' = \varnothing = \mathscr{A}$. Hence, since $\mathscr{A} = \varnothing$ and $\mathscr{R} \subseteq \mathscr{A} \times \mathscr{A}$, it follows that $\mathscr{R} = \varnothing$ and so $\mathcal{A} = (\varnothing, \varnothing)$. Thus, the "only if" is satisfied.

   Assume now that $\mathcal{A} = (\varnothing, \varnothing)$. Consider the empty graph $(\varnothing, \varnothing)$. We must prove that it is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$. First and second condition of Def. 39 follow immediately from $\mathscr{A}' = \mathscr{R}' = \varnothing$. The third condition follows from $\mathscr{A} = \varnothing$. Thus, the "if" is satisfied.

□

Theorem. 7 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{$Coh, Def, Rein1, Rein2, CA$\}$ be an Abstract Argumentation principle. If $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{π} for S on $\mathcal{A}$, then it is unique.*

Proof. (for Theorem 7) Assume that $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{π} for S on $\mathcal{A}$. Let (A', R') be a subgraph of $\mathcal{A}$. Suppose $(\mathscr{A}', \mathscr{R}')$ is also an answer to Q^{Ext}_{π} for S on $\mathcal{A}$. We must prove that $(\mathscr{A}', \mathscr{R}') = (\varnothing, \varnothing)$. Since $\mathscr{R}' \subseteq \mathscr{A}' \times \mathscr{A}'$, this can be reduced to proving that $\mathscr{A}' = \varnothing$.

Consider $\pi = \mathit{Coh}$ or Def or $\mathit{Rein2}$. Following Th. 6 and since $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{π} for S on $\mathcal{A}$, $S = \varnothing$ and so $\mathscr{R}^{-1}(S) = \mathscr{R}^{+2}(S) = \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) = \varnothing$. So following Condition 1 in Def. 34 (resp. Def. 36, Def. 39), $\mathscr{A}' = \varnothing$.

Consider $\pi = \mathit{Rein1}$ and let $X = \{a \in A \mid \mathscr{R}^{-1}(a) = \varnothing\}$. By assumption $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$, thus by Th. 6, we know that $X = \varnothing$, so by Condition 1 in Def. 38, $\mathscr{A}' = \varnothing$.

Consider $\pi = \mathit{CA}$. By assumption $(\varnothing, \varnothing)$ is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$, thus by Th. 6, we know that $\mathscr{A} = \varnothing$. Since $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $\mathcal{A}$, we deduce that $\mathscr{A}' = \varnothing$. □

A.2.2 Maximal and Minimal Explanations

Uniqueness of Maximal Explanations

Theorem. 8 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{$Coh, Def, Rein1, Rein2, CA$\}$ be an Abstract Argumentation principle. If $Expl_{\pi}(S)$ is a maximal explanation for π, then it is the*

unique maximal explanation for π.

The proof of Th. 8 relies on Lem. 6 to 10.

Proof. (for Theorem 8) Denote $Expl_\pi(S) = (\mathscr{A}', \mathscr{R}')$, assume $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for π and let $(\mathscr{A}'', \mathscr{R}'')$ be a subgraph of $\mathcal{A}$. Suppose $(\mathscr{A}'', \mathscr{R}'')$ is also a maximal explanation for π. We must prove that $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$.

Consider $\pi = Coh$ and let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Coh*, we know by Def. 34 that $\mathscr{A}'' = S$ and $\mathscr{R}'' \subseteq X$. But $(\mathscr{A}'', \mathscr{R}'')$ is a maximal explanation for *Coh*. Thus, by Lem. 6, $X \subseteq \mathscr{R}''$ and so $\mathscr{R}'' = X$. However, $(\mathscr{A}', \mathscr{R}')$ is also an explanation for *Coh*, so $\mathscr{A}' = S$ and $\mathscr{R}' \subseteq X$. Thus, $\mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}' \subseteq \mathscr{R}''$, so we conclude that $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $(\mathscr{A}'', \mathscr{R}'')$. If $\mathscr{R}' \subset \mathscr{R}''$, we conclude further that $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$, a contradiction with the assumption that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Coh*. So it must be the case that $\mathscr{R}' = \mathscr{R}''$.

Consider $\pi = Def$ and let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Def*, we know by Def. 36 that $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S)$, $X \subseteq \mathscr{R}''$ and $\mathscr{R}'' \subseteq X \cup Y$. But $(\mathscr{A}'', \mathscr{R}'')$ is a maximal explanation for *Def*. Thus, by Lem. 7, $Y \subseteq \mathscr{R}''$, so $X \cup Y \subseteq \mathscr{R}''$, and so $\mathscr{R}'' = X \cup Y$. However, $(\mathscr{A}', \mathscr{R}')$ is also an explanation for *Def*, so $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$ and $\mathscr{R}' \subseteq X \cup Y$. Thus, $\mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}' \subseteq \mathscr{R}''$, so we conclude that $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $(\mathscr{A}'', \mathscr{R}'')$. If $\mathscr{R}' \subset \mathscr{R}''$, we conclude further that $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$, a contradiction with the assumption that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Def*. So it must be the case that $\mathscr{R}' = \mathscr{R}''$.

Consider $\pi = Rein1$ and let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein1*, we know by Def. 38 that $\mathscr{A}'' \subseteq X$ and $\mathscr{R}'' = \varnothing$. But $(\mathscr{A}'', \mathscr{R}'')$ is a maximal explanation for *Rein1*. Thus, by Lem. 8, $X \subseteq \mathscr{A}''$, and so $\mathscr{A}'' = X$. However, $(\mathscr{A}', \mathscr{R}')$ is also an explanation for *Rein1*, so $\mathscr{A}' \subseteq X$ and $\mathscr{R}' = \varnothing$. Thus, $\mathscr{A}' \subseteq \mathscr{A}''$ and $\mathscr{R}' = \mathscr{R}''$, so we conclude that $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $(\mathscr{A}'', \mathscr{R}'')$. If $\mathscr{A}' \subset \mathscr{A}''$, we conclude further that $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$, a contradiction with the assumption that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein1*. So it must be the case that $\mathscr{A}' = \mathscr{A}''$.

Consider $\pi = Rein2$ and let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein2*, we know by Def. 39 that $\mathscr{A}'' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $X \subseteq \mathscr{R}''$, and $\mathscr{R}'' \subseteq X \cup Y$. But $(\mathscr{A}'', \mathscr{R}'')$ is a maximal explanation for *Rein2*. Thus, by Lem. 9, $Y \subseteq \mathscr{R}''$, so $X \cup Y \subseteq \mathscr{R}''$, and so $\mathscr{R}'' = X \cup Y$. However, $(\mathscr{A}', \mathscr{R}')$ is also an explanation for *Rein2*, so $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $\mathscr{R}' \subseteq X \cup Y$. Thus, $\mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}' \subseteq \mathscr{R}''$, so we conclude that $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $(\mathscr{A}'', \mathscr{R}'')$. If $\mathscr{R}' \subset \mathscr{R}''$, we conclude further that $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$, a contradiction with the assumption that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein2*. So it must be the case that $\mathscr{R}' = \mathscr{R}''$.

Consider $\pi = CA$ and let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *CA*, we know by Def. 41 that $\mathscr{A}'' = \mathscr{A}$ and $\mathscr{R}'' \subseteq X$. But $(\mathscr{A}'', \mathscr{R}'')$ is a maximal explanation for *CA*. Thus, by Lem. 10, $X \subseteq \mathscr{R}''$, and so $\mathscr{R}'' = X$. However, $(\mathscr{A}', \mathscr{R}')$ is also an explanation for *CA*, so $\mathscr{A}' = \mathscr{A}$ and $\mathscr{R}' \subseteq X$. Thus, $\mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}' \subseteq \mathscr{R}''$, so we conclude that $(\mathscr{A}', \mathscr{R}')$ is a subgraph of $(\mathscr{A}'', \mathscr{R}'')$. If $\mathscr{R}' \subset \mathscr{R}''$, we conclude further that $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$, a contradiction with the assumption that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *CA*. So it must be the case that $\mathscr{R}' = \mathscr{R}''$. □

Characterization of Minimal Explanations

Lemma. 1 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Coh. $Expl_{Coh}(S)$ is a minimal explanation for Coh if and only if $|\mathscr{R}'| \leq 1$.*

Proof. (for Lemma 1) Suppose that $|\mathscr{R}'| > 1$. Let $(x, y) \in \mathscr{R}'$ and consider $(\mathscr{A}', \mathscr{R}'')$ such that $\mathscr{R}'' = \mathscr{R}' \setminus \{(x, y)\}$. Obviously, $(\mathscr{A}', \mathscr{R}'')$ is a strict subgraph of $(\mathscr{A}', \mathscr{R}')$. Let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Coh*, by Def. 34, we know that $\mathscr{A}' = S$, $\mathscr{R}' \subseteq X$, and because $|\mathscr{R}'| > 1$, $X \neq \varnothing$. However, $\mathscr{R}'' \subset \mathscr{R}'$, so $\mathscr{R}'' \subset X$ and $|\mathscr{R}''| \geq 1$, so $\mathscr{R}'' \neq \varnothing$. Thus, by Def. 34, $(\mathscr{A}', \mathscr{R}'')$ is an explanation for *Coh*, which contradicts the minimality of $(\mathscr{A}', \mathscr{R}')$.

Suppose now that $|\mathscr{R}'| \leq 1$ and there exists a strict subgraph $(\mathscr{A}'', \mathscr{R}'')$ of $(\mathscr{A}', \mathscr{R}')$ such that $(\mathscr{A}'', \mathscr{R}'')$ is also an explanation for *Coh*. Let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. Assume firstly that $\mathscr{A}'' \subset \mathscr{A}'$. By Def. 34,

we know that $\mathscr{A}'' = S$, so we have $S \subset \mathscr{A}'$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Coh*. Assume secondly that $\mathscr{R}'' \subset \mathscr{R}'$. By supposition $|\mathscr{R}'| \leq 1$, so in this case, $|\mathscr{R}'| = 1$ and $\mathscr{R}'' = \varnothing$. As such, $\mathscr{R}'' \subseteq X$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Coh* and $\mathscr{R}'' = \varnothing$, by Def. 34, $X = \varnothing$. However, as $|\mathscr{R}'| = 1$, this would mean that $\mathscr{R}' \not\subseteq X$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Coh*. Consequently, it must be the case that $\mathscr{R}'' = \mathscr{R}'$, and so $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}', \mathscr{R}')$ is a minimal explanation for *Coh*. □

Lemma. 2 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Def. $Expl_{Def}(S)$ is a minimal explanation for Def if and only if for all $x \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Proof. (for Lemma 2) Suppose that there exists $x \in \mathscr{R}^{-1}(S) \setminus S$ such that $|\mathscr{R}'^{-1}(x)| > 1$. Let $(w, x) \in \mathscr{R}'$ and consider $(\mathscr{A}', \mathscr{R}'')$ such that $\mathscr{R}'' = \mathscr{R}' \setminus \{(w, x)\}$. Obviously, $(\mathscr{A}', \mathscr{R}'')$ is a strict subgraph of $(\mathscr{A}', \mathscr{R}')$. Let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Def*, by Def. 36, we know that $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$, $X \subseteq \mathscr{R}' \subseteq X \cup Y$. In addition, because $x \in \mathscr{R}^{-1}(S)$ and $x \notin S$, it cannot be that $(w, x) \in X$, so $(w, x) \in (Y \setminus X)$ and $x \in \mathscr{R}^{+1}(S)$. However, $\mathscr{R}'' \subset \mathscr{R}'$, so $\mathscr{R}'' \subset X \cup Y$ and since $(w, x) \in (Y \setminus X)$, $X \subseteq \mathscr{R}''$. Moreover, as $|\mathscr{R}'^{-1}(x)| > 1$, we have that $|\mathscr{R}''^{-1}(x)| \geq 1$, so $\exists (w', x) \in \mathscr{R}''$; consequently, knowing that $x \in \mathscr{R}^{-1}(S)$ and $x \in \mathscr{R}^{+1}(S)$, the third condition of Def. 36 is satisfied for each $b \in \mathscr{R}^{-1}(S)$ including x. Thus, by Def. 36, $(\mathscr{A}', \mathscr{R}'')$ is an explanation for *Def*, which contradicts the minimality of $(\mathscr{A}', \mathscr{R}')$.

Suppose now that for all $x \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}'^{-1}(x)| \leq 1$ and that there exists a strict subgraph $(\mathscr{A}'', \mathscr{R}'')$ of $(\mathscr{A}', \mathscr{R}')$ such that $(\mathscr{A}'', \mathscr{R}'')$ is also an explanation for *Def*. Let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. Assume firstly that $\mathscr{A}'' \subset \mathscr{A}'$. By Def. 36, we know that $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S)$, so we have $S \cup \mathscr{R}^{-1}(S) \subset \mathscr{A}'$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Def*. Assume secondly that $\mathscr{R}'' \subset \mathscr{R}'$. By Def. 36, we have $X \subseteq \mathscr{R}' \subseteq X \cup Y$ and $X \subseteq \mathscr{R}'' \subseteq X \cup Y$; so since $\mathscr{R}'' \subset \mathscr{R}'$ there exists at least $(a, b) \in \mathscr{R}' \setminus \mathscr{R}''$ and by definition $(a, b) \in Y \setminus X$, so $b \in \mathscr{R}^{-1}(S) \setminus S$. Moreover, by supposition, for all $x \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}'^{-1}(x)| \leq 1$, so in this case, we have $|\mathscr{R}'^{-1}(b)| = 1$ and $\mathscr{R}''^{-1}(b) = \varnothing$. In addition, as $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Def*, $b \in \mathscr{R}^{-1}(S)$ and $\mathscr{R}''^{-1}(b) = \varnothing$, by the third condition of Def. 36, $b \notin \mathscr{R}^{+1}(S)$ and so $(a, b) \notin Y$. This would mean that $\mathscr{R}' \not\subseteq X \cup Y$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Def*. Consequently, it must be the case that $\mathscr{R}'' = \mathscr{R}'$, and so $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}', \mathscr{R}')$ is a minimal explanation for *Def*. □

Lemma. 3 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Rein1. $Expl_{Rein1}(S)$ is a minimal explanation for Rein1 if and only if $|\mathscr{A}' \setminus S| \leq 1$.*

Proof. (for Lemma 3) Suppose that $|\mathscr{A}' \setminus S| > 1$. Let $x \in \mathscr{A}' \setminus S$ and consider $(\mathscr{A}'', \mathscr{R}')$ such that $\mathscr{A}'' = \mathscr{A}' \setminus \{x\}$. Obviously, $(\mathscr{A}'', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}', \mathscr{R}')$. Let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*, by Def. 38, we know that $\mathscr{R}' = \varnothing$, $S \cap X \subseteq \mathscr{A}' \subseteq X$, and because $x \notin S$, $x \in X \setminus S$. However, $\mathscr{A}'' \subset \mathscr{A}'$, so $\mathscr{A}'' \subset X$ and since $x \in \mathscr{A}' \setminus S$, $S \cap X \subseteq \mathscr{A}''$. Moreover, as $|\mathscr{A}' \setminus S| > 1$ and $x \in X$, we have that $|(\mathscr{A}'' \setminus S) \cap X| \geq 1$. Thus, by Def. 38, $(\mathscr{A}'', \mathscr{R}')$ is an explanation for *Rein1*, which contradicts the minimality of $(\mathscr{A}', \mathscr{R}')$.

Suppose now that $|\mathscr{A}' \setminus S| \leq 1$ and that there exists a strict subgraph $(\mathscr{A}'', \mathscr{R}'')$ of $(\mathscr{A}', \mathscr{R}')$ such that $(\mathscr{A}'', \mathscr{R}'')$ is also an explanation for *Rein1*. Let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. Assume firstly that $\mathscr{R}'' \subset \mathscr{R}'$. By Def. 38, we know that $\mathscr{R}'' = \varnothing$, so we have $|\mathscr{R}'| > 0$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*. Assume secondly that $\mathscr{A}'' \subset \mathscr{A}'$. As $(\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}'', \mathscr{R}'')$ are explanations for *Rein1*, by Def. 38, $S \cap X \subseteq \mathscr{A}' \subseteq X$ and $S \cap X \subseteq \mathscr{A}'' \subseteq X$. Since $\mathscr{A}'' \subset \mathscr{A}'$, we have that $\mathscr{A}'' \setminus S \subset \mathscr{A}' \setminus S$. By supposition $|\mathscr{A}' \setminus S| \leq 1$, so in this case, $|\mathscr{A}' \setminus S| = 1$ and $\mathscr{A}'' \setminus S = \varnothing$. As $\mathscr{A}'' \setminus S = \varnothing$, $\nexists a \in (\mathscr{A} \setminus S) \cap X$ with $a \in \mathscr{A}''$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein1*, by Def. 38, $(\mathscr{A} \setminus S) \cap X = \varnothing$. However, as $|\mathscr{A}' \setminus S| = 1$, this would mean that $\mathscr{A}' \not\subseteq X$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*. Consequently, it must be the case that $\mathscr{A}'' = \mathscr{A}'$, and so $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}', \mathscr{R}')$ is a minimal explanation for *Rein1*. □

Lemma. 4 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{Rein2}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for Rein2. $Expl_{Rein2}(S)$ is a minimal explanation for Rein2 if and only if for all $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Proof. (for Lemma 4) Suppose that there exists $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$ such that $|\mathscr{R}'^{-1}(x)| > 1$. Let $(w, x) \in \mathscr{R}'$ and consider $(\mathscr{A}', \mathscr{R}'')$ such that $\mathscr{R}'' = \mathscr{R}' \setminus \{(w, x)\}$. Obviously, $(\mathscr{A}', \mathscr{R}'')$ is a strict subgraph of $(\mathscr{A}', \mathscr{R}')$. Let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein2*, by Def. 39, we know that $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $X \subseteq \mathscr{R}' \subseteq X \cup Y$. In addition, because $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $x \notin \mathscr{R}^{+2}(S)$, it cannot be that $(w, x) \in X$, so $(w, x) \in (Y \setminus X)$ and $x \in \mathscr{R}^{+1}(S)$. However, $\mathscr{R}'' \subset \mathscr{R}'$, so $\mathscr{R}'' \subset X \cup Y$ and since $(w, x) \in (Y \setminus X)$, $X \subseteq \mathscr{R}''$. Moreover, as $|\mathscr{R}'^{-1}(x)| > 1$, we have that $|\mathscr{R}''^{-1}(x)| \geq 1$, so $\exists (w', x) \in \mathscr{R}''$; consequently, knowing that $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $x \in \mathscr{R}^{+1}(S)$, the third condition of Def. 39 is satisfied for each $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ including x. Thus, by Def. 39, $(\mathscr{A}', \mathscr{R}'')$ is an explanation for *Rein2*, which contradicts the minimality of $(\mathscr{A}', \mathscr{R}')$.

Suppose now that for all $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}'^{-1}(x)| \leq 1$ and that there exists a strict subgraph $(\mathscr{A}'', \mathscr{R}'')$ of $(\mathscr{A}', \mathscr{R}')$ such that $(\mathscr{A}'', \mathscr{R}'')$ is also an explanation for *Rein2*. Let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. Assume firstly that $\mathscr{A}'' \subset \mathscr{A}'$. By Def. 39, we know that $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S)$, so we have $S \cup \mathscr{R}^{-1}(S) \subset \mathscr{A}'$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein2*. Assume secondly that $\mathscr{R}'' \subset \mathscr{R}'$. By Def. 39, we have $X \subseteq \mathscr{R}' \subseteq X \cup Y$ and $X \subseteq \mathscr{R}'' \subseteq X \cup Y$; so since $\mathscr{R}'' \subset \mathscr{R}'$ there exists at least $(a, b) \in \mathscr{R}' \setminus \mathscr{R}''$ and by definition $(a, b) \in Y \setminus X$, so $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$. Moreover, by supposition, for all $x \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}'^{-1}(x)| \leq 1$, so in this case, we have $|\mathscr{R}'^{-1}(b)| = 1$ and $\mathscr{R}''^{-1}(b) = \varnothing$. In addition, as $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein2*, $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $\mathscr{R}''^{-1}(b) = \varnothing$, by the third condition of Def. 39, $b \notin \mathscr{R}^{+1}(S)$ and so $(a, b) \notin Y$. This would mean that $\mathscr{R}' \not\subseteq X \cup Y$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein2*. Consequently, it must be the case that $\mathscr{R}'' = \mathscr{R}'$, and so $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}', \mathscr{R}')$ is a minimal explanation for *Rein2*. □

Lemma. 5 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ be an explanation for CA. $Expl_{CA}(S)$ is a minimal explanation for CA if and only if for all $x \notin S$, $|\mathscr{R}'^{-1}(x)| \leq 1$.*

Proof. (for Lemma 5) Suppose that there exists $x \notin S$ such that $|\mathscr{R}'^{-1}(x)| > 1$. Let $(w, x) \in \mathscr{R}'$ and consider $(\mathscr{A}', \mathscr{R}'')$ such that $\mathscr{R}'' = \mathscr{R}' \setminus \{(w, x)\}$. Obviously, $(\mathscr{A}', \mathscr{R}'')$ is a strict subgraph of $(\mathscr{A}', \mathscr{R}')$. Let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *CA*, by Def. 41, we know that $\mathscr{A}' = \mathscr{A}$, $\mathscr{R}' \subseteq X$, and because $x \notin S$, $(w, x) \in X$ so $x \in \mathscr{R}^{+1}(S)$. However, $\mathscr{R}'' \subset \mathscr{R}'$, so $\mathscr{R}'' \subset X$ and as $|\mathscr{R}'^{-1}(x)| > 1$, we have that $|\mathscr{R}''^{-1}(x)| \geq 1$, so $\exists (w', x) \in \mathscr{R}''$ and because $\mathscr{R}'' \subset X$ and $x \notin S$, $w' \in S$. Thus, by Def. 41, $(\mathscr{A}', \mathscr{R}'')$ is an explanation for *CA*, which contradicts the minimality of $(\mathscr{A}', \mathscr{R}')$.

Suppose now that for all $x \notin S$, $|\mathscr{R}'^{-1}(x)| \leq 1$ and that there exists a strict subgraph $(\mathscr{A}'', \mathscr{R}'')$ of $(\mathscr{A}', \mathscr{R}')$ such that $(\mathscr{A}'', \mathscr{R}'')$ is also an explanation for *CA*. Let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. Assume firstly that $\mathscr{A}'' \subset \mathscr{A}'$. By Def. 41, we know that $\mathscr{A}'' = \mathscr{A}$, so we have $\mathscr{A} \subset \mathscr{A}'$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *CA*. Assume secondly that $\mathscr{R}'' \subset \mathscr{R}'$. By Def. 41, we have $\mathscr{R}' \subseteq X$ and by supposition, for all $x \notin S$, $|\mathscr{R}'^{-1}(x)| \leq 1$, so in this case, there exists $b \notin S$ such that $|\mathscr{R}'^{-1}(b)| = 1$ and $\mathscr{R}''^{-1}(b) = \varnothing$. As such, $\mathscr{R}'' \subseteq X$. Since $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *CA*, $b \notin S$ and $\mathscr{R}''^{-1}(b) = \varnothing$, by Def. 41, $b \notin \mathscr{R}^{+1}(S)$. However, as $|\mathscr{R}'^{-1}(b)| = 1$, this would mean that $\mathscr{R}' \not\subseteq X$, which contradicts the fact that $(\mathscr{A}', \mathscr{R}')$ is an explanation for *CA*. Consequently, it must be the case that $\mathscr{R}'' = \mathscr{R}'$, and so $(\mathscr{A}'', \mathscr{R}'') = (\mathscr{A}', \mathscr{R}')$ and $(\mathscr{A}', \mathscr{R}')$ is a minimal explanation for *CA*. □

Equality between Maximal Explanation and Union of all Minimal Explanations

Theorem. 9 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{$Coh, Def, Rein1, Rein2, CA$\}$ be an Abstract Argumentation principle. Consider a maximal explanation for π $MaxExpl_\pi(S)$, and let M be the set of all minimal explanations for π. Then $Expl_\pi(S) = \bigcup_{G \in M} G$.*

The proof of Th. 9 relies on Lem. 6 to 10.

Proof. (for Theorem 9) • Denote $MaxExpl_\pi(S) = (\mathscr{A}', \mathscr{R}')$ and $M = \{G_1, \ldots, G_n\}$ with $G_1 = (\mathscr{A}_1, \mathscr{R}_1)$, …, $G_n = (\mathscr{A}_n, \mathscr{R}_n)$. We prove that $(\mathscr{A}', \mathscr{R}') \subseteq \cup_{G \in M} G$. Suppose that $(\mathscr{A}', \mathscr{R}') \not\subseteq \bigcup_{G \in M} G$. This means that $\mathscr{A}' \not\subseteq \mathscr{A}_1 \cup \cdots \cup \mathscr{A}_n$ or $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \cdots \cup \mathscr{R}_n$.

Consider $\pi = Coh$ and let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. By supposition, $(\mathscr{A}', \mathscr{R}'), G_1, \ldots, G_n$ are all explanations for *Coh*. So, by Def. 34, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S$. Thus, it must be the case that $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \cdots \cup \mathscr{R}_n$. In addition, by Lem. 6, we know that $\mathscr{R}' = X$ and so that $\mathscr{R}_1 \subseteq \mathscr{R}'$, …, $\mathscr{R}_n \subseteq \mathscr{R}'$. Assume firstly that $X = \varnothing$. Then, by Def. 34, we have $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = \varnothing$, a contradiction. Assume secondly that $X \neq \varnothing$. In this case we have $\mathscr{R}' \neq \varnothing$, and by Def. 34, $\mathscr{R}_1 \neq \varnothing$, …, $\mathscr{R}_n \neq \varnothing$. This means that there exists $\mathscr{R}'' \subseteq X$ with $\mathscr{R}'' \neq \varnothing$, such that $\mathscr{R}'' \cap \mathscr{R}_1 = \cdots = \mathscr{R}'' \cap \mathscr{R}_n = \varnothing$. Let $\mathscr{R}''' \subseteq \mathscr{R}''$ with $|\mathscr{R}'''| = 1$. Consider $(\mathscr{A}', \mathscr{R}''')$. We already know that $\mathscr{A}' = S$, $\mathscr{R}''' \subseteq X$ and we have both $X \neq \varnothing$ and $\mathscr{R}''' \neq \varnothing$. So, by Def. 34, $(\mathscr{A}', \mathscr{R}''')$ is an explanation for *Coh*. In addition, since $\mathscr{R}'' \cap \mathscr{R}_1 = \cdots = \mathscr{R}'' \cap \mathscr{R}_n = \varnothing$, we have $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_1, \mathscr{R}_1)$, …, $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_n, \mathscr{R}_n)$. However, $|\mathscr{R}'''| = 1$, so by Lem. 1, $(\mathscr{A}', \mathscr{R}''')$ is a minimal explanation for *Coh*. A contradiction with the hypothesis that M is the set of all minimal explanations for *Coh*.

Consider $\pi = Def$ and let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. By supposition, $(\mathscr{A}', \mathscr{R}'), G_1, \ldots, G_n$ are all explanations for *Def*. So, by Def. 34, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S \cup \mathscr{R}^{-1}(S)$. Thus, it must be the case that $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \cdots \cup \mathscr{R}_n$. In addition, by Lem. 7, we know that $\mathscr{R}' = X \cup Y$ and so that $\mathscr{R}_1 \subseteq \mathscr{R}'$, …, $\mathscr{R}_n \subseteq \mathscr{R}'$. Assume firstly that for all $y \in \mathscr{R}^{-1}(S)$, $\mathscr{R}'^{-1}(y) = \varnothing$. Then, by Def. 36, we have for all $y \in \mathscr{R}^{-1}(S)$, $y \notin \mathscr{R}^{+1}(S)$. Since $X \subseteq \mathscr{R}_1 \subseteq X \cup Y$, …, $X \subseteq \mathscr{R}_n \subseteq X \cup Y$, we deduce that for all $y \in \mathscr{R}^{-1}(S)$, $\mathscr{R}_1^{-1}(y) = \cdots = \mathscr{R}_n^{-1}(y) = \mathscr{R}'^{-1}(y) = \varnothing$, and so that $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = X$, a contradiction. Assume secondly that for some $y \in \mathscr{R}^{-1}(S)$, $\mathscr{R}'^{-1}(y) \neq \varnothing$. In this case we have $\mathscr{R}' \neq \varnothing$ and $y \in \mathscr{R}^{+1}(S)$, so by Def. 36, $\mathscr{R}_1^{-1}(y) \neq \varnothing$, …, $\mathscr{R}_n^{-1}(y) \neq \varnothing$ and thus, $\mathscr{R}_1 \neq \varnothing$, …, $\mathscr{R}_n \neq \varnothing$. Since, $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \ldots \cup \mathscr{R}_n$ and $\forall i, \mathscr{R}_i \subseteq \mathscr{R}'$, this means that there exists $\mathscr{R}'' \subseteq Y$ with $\mathscr{R}'' \neq \varnothing$, such that $\mathscr{R}'' \cap \mathscr{R}_1 = \cdots = \mathscr{R}'' \cap \mathscr{R}_n = \varnothing$. In particular, this means that there exists $y_0 \in \mathscr{R}^{-1}(S)$ such that $\mathscr{R}''^{-1}(y_0) \neq \varnothing$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$. Moreover, because $X \subseteq \mathscr{R}_1$, …$X \subseteq \mathscr{R}_n$, we know that $y_0 \notin S$. Let $\mathscr{R}'''$ such that: (1) $X \subseteq \mathscr{R}''' \subset \mathscr{R}'$, (2) for all $y \in \mathscr{R}^{-1}(S) \setminus S, y \neq y_0$, $|\mathscr{R}'''^{-1}(y)| = 1$ if $y \in \mathscr{R}^{+1}(S)$ and $|\mathscr{R}'''^{-1}(y)| = 0$ otherwise, (3) $|\mathscr{R}'''^{-1}(y_0)| = 1$ with $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$. Consider $(\mathscr{A}', \mathscr{R}''')$. We already know that $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$ and $X \subseteq \mathscr{R}''' \subseteq X \cup Y$. In addition, we have that for all $y \in \mathscr{R}^{-1}(S)$ such that $y \in \mathscr{R}^{+1}(S)$, $\mathscr{R}'''^{-1}(y) \neq \varnothing$ (definition of $\mathscr{R}'''$). So, by Def. 36, $(\mathscr{A}', \mathscr{R}''')$ is an explanation for *Def*. In addition, since $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$, we have $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_1, \mathscr{R}_1)$, …, $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_n, \mathscr{R}_n)$. However, for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}'''-1(y)| \leq 1$, so by Lem. 2, $(\mathscr{A}', \mathscr{R}''')$ is a minimal explanation for *Def*. A contradiction with the hypothesis that M is the set of all minimal explanations for *Def*.

Consider $\pi = Rein1$ and let $X = \{a \in A \mid \mathscr{R}^{-1}(a) = \varnothing\}$. By supposition, $(\mathscr{A}', \mathscr{R}'), G_1, \ldots, G_n$ are all explanations for *Rein1*. So, by Def. 38, we have $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = \varnothing$. Thus, it must be the case that $\mathscr{A}' \not\subseteq \mathscr{A}_1 \cup \cdots \cup \mathscr{A}_n$. In addition, by Lem. 8, we know that $\mathscr{A}' = X$ and so that $\mathscr{A}_1 \subseteq \mathscr{A}'$, …, $\mathscr{A}_n \subseteq \mathscr{A}'$. Assume firstly that $X \setminus S = \varnothing$ (or, written differently, $S \cap X = X$ and $(\mathscr{A} \setminus S) \cap X = \varnothing$). Then, by Def. 38, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = X$, a contradiction. Assume secondly that $X \setminus S \neq \varnothing$, so $(\mathscr{A} \setminus S) \cap X \neq \varnothing$. By assumption $\mathscr{A}' \not\subseteq \mathscr{A}_1 \cup \cdots \cup \mathscr{A}_n$, so there exists $\mathscr{A}'' \subseteq X \setminus S$ with $\mathscr{A}'' \neq \varnothing$, such that $\mathscr{A}'' \cap \mathscr{A}_1 = \cdots = \mathscr{A}'' \cap \mathscr{A}_n = \varnothing$. Let $x \in \mathscr{A}''$ and $\mathscr{A}''' = (S \cap X) \cup \{x\}$. Consider $(\mathscr{A}''', \mathscr{R}')$. We already know that $\mathscr{R}' = \varnothing$, $S \cap X \subseteq \mathscr{A}''' \subseteq X$ and we have both $(\mathscr{A} \setminus S) \cap X \neq \varnothing$ and $x \in ((\mathscr{A} \setminus S) \cap X) \cap \mathscr{A}'''$. So, by Def. 38, $(\mathscr{A}''', \mathscr{R}')$ is an explanation for *Rein1*. In addition, since $\mathscr{A}'' \cap \mathscr{A}_1 = \cdots = \mathscr{A}'' \cap \mathscr{A}_n = \varnothing$, we have $(\mathscr{A}''', \mathscr{R}') \neq (\mathscr{A}_1, \mathscr{R}_1)$, …, $(\mathscr{A}''', \mathscr{R}') \neq (\mathscr{A}_n, \mathscr{R}_n)$. However, $|\mathscr{A}''' \setminus S| = 1$, so by Lem. 3, $(\mathscr{A}''', \mathscr{R}')$ is a minimal explanation for *Rein1*. A contradiction with the hypothesis that M is the set of all minimal explanations for *Rein1*.

Consider $\pi = Rein2$ and let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. By supposition, $(\mathscr{A}', \mathscr{R}'), G_1, \ldots, G_n$ are all explanations for *Rein2*. So, by Def. 39, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. Thus, it must be the case that $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \cdots \cup \mathscr{R}_n$. In addition, by Lem. 9, we know that $\mathscr{R}' = X \cup Y$ and so that $\mathscr{R}_1 \subseteq \mathscr{R}'$, …, $\mathscr{R}_n \subseteq \mathscr{R}'$. Assume firstly that for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $\mathscr{R}'^{-1}(y) = \varnothing$. Then, by Def. 39, we have for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $y \notin \mathscr{R}^{+1}(S)$ (so $Y = \varnothing$). Since $X \subseteq \mathscr{R}_1 \subseteq X \cup Y$, …, $X \subseteq \mathscr{R}_n \subseteq X \cup Y$, we

deduce that $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = X$, a contradiction. Assume secondly that for some $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $\mathscr{R}'^{-1}(y) \neq \varnothing$. In this case we have $\mathscr{R}' \neq \varnothing$ and $y \in \mathscr{R}^{+1}(S)$, so by Def. 39, $\mathscr{R}_1^{-1}(y) \neq \varnothing, \ldots, \mathscr{R}_n^{-1}(y) \neq \varnothing$ and thus, $\mathscr{R}_1 \neq \varnothing, \ldots, \mathscr{R}_n \neq \varnothing$. Since, $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \ldots \cup \mathscr{R}_n$ and $\forall i, \mathscr{R}_i \subseteq \mathscr{R}'$, this means that there exists $\mathscr{R}'' \subseteq Y$ with $\mathscr{R}'' \neq \varnothing$, such that $\mathscr{R}'' \cap \mathscr{R}_1 = \cdots = \mathscr{R}'' \cap \mathscr{R}_n = \varnothing$. In particular, this means that there exists $y_0 \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ such that $\mathscr{R}''^{-1}(y_0) \neq \varnothing$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$. Moreover, because $X \subseteq \mathscr{R}_1, \ldots X \subseteq \mathscr{R}_n$, we know that $y_0 \notin \mathscr{R}^{+2}(S)$. Let $\mathscr{R}'''$ such that: (1) $X \subseteq \mathscr{R}''' \subset \mathscr{R}'$, (2) for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S), y \neq y_0$, $|\mathscr{R}'''^{-1}(y)| = 1$ if $y \in \mathscr{R}^{+1}(S)$ and $|\mathscr{R}'''^{-1}(y)| = 0$ otherwise, (3) $|\mathscr{R}'''^{-1}(y_0)| = 1$ and $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$. Consider $(\mathscr{A}', \mathscr{R}''')$. We already know that $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $X \subseteq \mathscr{R}''' \subseteq X \cup Y$. In addition, we have that for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ such that $y \in \mathscr{R}^{+1}(S)$, $\mathscr{R}'''^{-1}(y) \neq \varnothing$ (definition of $\mathscr{R}'''$). So, by Def. 39, $(\mathscr{A}', \mathscr{R}''')$ is an explanation for *Rein2*. In addition, since $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$, we have $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_1, \mathscr{R}_1), \ldots, (\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_n, \mathscr{R}_n)$. However, for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}'''^{-1}(y)| \leq 1$, so by Lem. 4, $(\mathscr{A}', \mathscr{R}''')$ is a minimal explanation for *Rein2*. A contradiction with the hypothesis that M is the set of all minimal explanations for *Rein2*.

Consider $\pi = CA$ and let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. By supposition, $(\mathscr{A}', \mathscr{R}'), G_1, \ldots, G_n$ are all explanations for *CA*. So, by Def. 41, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = \mathscr{A}$. Thus, it must be the case that $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \cdots \cup \mathscr{R}_n$. In addition, by Lem. 10, we know that $\mathscr{R}' = X$ and so that $\mathscr{R}_1 \subseteq \mathscr{R}', \ldots, \mathscr{R}_n \subseteq \mathscr{R}'$. Assume firstly that for all $y \notin S$, $\mathscr{R}'^{-1}(y) = \varnothing$. Then, by Def. 41, we have for all $y \notin S$, $y \notin \mathscr{R}^{+1}(S)$ (so $X = \varnothing$). Since $\mathscr{R}_1 \subseteq X, \ldots, \mathscr{R}_n \subseteq X$, we deduce that $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = \varnothing$, a contradiction. Assume secondly that for some $y \notin S$, $\mathscr{R}'^{-1}(y) \neq \varnothing$. In this case we have $\mathscr{R}' \neq \varnothing$ and $y \in \mathscr{R}^{+1}(S)$, so by Def. 41, $\mathscr{R}_1^{-1}(y) \neq \varnothing, \ldots, \mathscr{R}_n^{-1}(y) \neq \varnothing$ and thus, $\mathscr{R}_1 \neq \varnothing, \ldots, \mathscr{R}_n \neq \varnothing$. Since, by assumption, $\mathscr{R}' \not\subseteq \mathscr{R}_1 \cup \ldots \cup \mathscr{R}_n$, there exists $\mathscr{R}'' \subseteq X$ with $\mathscr{R}'' \neq \varnothing$, such that $\mathscr{R}'' \cap \mathscr{R}_1 = \cdots = \mathscr{R}'' \cap \mathscr{R}_n = \varnothing$. In particular, this means that there exists $y_0 \notin S$ such that $\mathscr{R}''^{-1}(y_0) \neq \varnothing$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$. Let $\mathscr{R}''' \subset \mathscr{R}'$ such that: (1) for all $y \notin S, y \neq y_0$, $|\mathscr{R}'''^{-1}(y)| \leq 1$ if $y \in \mathscr{R}^{+1}(S)$ and $|\mathscr{R}'''^{-1}(y)| = 1$ otherwise, (2) $|\mathscr{R}'''^{-1}(y_0)| = 1$ with $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$. Consider $(\mathscr{A}', \mathscr{R}''')$. We already know that $\mathscr{A}' = \mathscr{A}$, $\mathscr{R}''' \subseteq X$ and for all $y \notin S$ such that $y \in \mathscr{R}^{+1}(S)$, we have $\mathscr{R}'''-1(y) \neq \varnothing$. So, by Def. 41, $(\mathscr{A}', \mathscr{R}''')$ is an explanation for *CA*. In addition, since $\mathscr{R}'''^{-1}(y_0) \subseteq \mathscr{R}''^{-1}(y_0)$ and $\mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_1^{-1}(y_0) = \cdots = \mathscr{R}''^{-1}(y_0) \cap \mathscr{R}_n^{-1}(y_0) = \varnothing$, we have $(\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_1, \mathscr{R}_1), \ldots, (\mathscr{A}', \mathscr{R}''') \neq (\mathscr{A}_n, \mathscr{R}_n)$. However, for all $y \notin S$, $|\mathscr{R}'''-1(y)| \leq 1$, so by Lem. 5, $(\mathscr{A}', \mathscr{R}''')$ is a minimal explanation for *CA*. A contradiction with the hypothesis that M is the set of all minimal explanations for *CA*.

• Denote $MaxExpl_\pi(S) = (\mathscr{A}', \mathscr{R}')$ and $M = \{G_1, \ldots, G_n\}$ with $G_1 = (\mathscr{A}_1, \mathscr{R}_1), \ldots, G_n = (\mathscr{A}_n, \mathscr{R}_n)$. We prove $(\mathscr{A}', \mathscr{R}') \supseteq \cup_{G \in M} G$.

Consider $\pi = Coh$ and let $X = \{(a, b) \in \mathscr{R} \mid a, b \in S\}$. By Def. 34, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S$. Moreover, by Lem. 6, we know that $\mathscr{R}' = X$. Finally, by Def. 34, we know that $\mathscr{R}_1 \subseteq X, \ldots, \mathscr{R}_n \subseteq X$. Thus, $\mathscr{R}_1 \subseteq \mathscr{R}', \ldots, \mathscr{R}_n \subseteq \mathscr{R}'$ and so $\bigcup_{G \in M} G \subseteq (\mathscr{A}', \mathscr{R}')$.

Consider $\pi = Def$ and let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. By Def. 36, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S \cup \mathscr{R}^{-1}(S)$. Moreover, by Lem. 7, we know that $\mathscr{R}' = X \cup Y$. Finally, by Def. 36, we know that $\mathscr{R}_1 \subseteq X \cup Y, \ldots, \mathscr{R}_n \subseteq X \cup Y$. Thus, $\mathscr{R}_1 \subseteq \mathscr{R}', \ldots, \mathscr{R}_n \subseteq \mathscr{R}'$ and so $\bigcup_{G \in M} G \subseteq (\mathscr{A}', \mathscr{R}')$.

Consider $\pi = Rein1$ and let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. By Def. 38, we have $\mathscr{R}' = \mathscr{R}_1 = \cdots = \mathscr{R}_n = \varnothing$. Moreover, by Lem. 8, we know that $\mathscr{A}' = X$. Finally, by Def. 38, we know that $\mathscr{A}_1 \subseteq X, \ldots, \mathscr{A}_n \subseteq X$. Thus, $\mathscr{A}_1 \subseteq \mathscr{A}', \ldots, \mathscr{A}_n \subseteq \mathscr{A}'$ and so $\bigcup_{G \in M} G \subseteq (\mathscr{A}', \mathscr{R}')$.

Consider $\pi = Rein2$ and let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. By Def. 39, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. Moreover, by Lem. 9, we know that $\mathscr{R}' = X \cup Y$. Finally, by Def. 39, we know that $\mathscr{R}_1 \subseteq X \cup Y, \ldots, \mathscr{R}_n \subseteq X \cup Y$. Thus, $\mathscr{R}_1 \subseteq \mathscr{R}', \ldots, \mathscr{R}_n \subseteq \mathscr{R}'$ and so $\bigcup_{G \in M} G \subseteq (\mathscr{A}', \mathscr{R}')$.

Consider $\pi = CA$ and let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. By Def. 41, we have $\mathscr{A}' = \mathscr{A}_1 = \cdots = \mathscr{A}_n = \mathscr{A}$. Moreover, by Lem. 10, we know that $\mathscr{R}' = X$. Finally, by Def. 41, we know that $\mathscr{R}_1 \subseteq X, \ldots, \mathscr{R}_n \subseteq X$. Thus, $\mathscr{R}_1 \subseteq \mathscr{R}', \ldots, \mathscr{R}_n \subseteq \mathscr{R}'$ and so $\bigcup_{G \in M} G \subseteq (\mathscr{A}', \mathscr{R}')$. □

A.3 Computation of Explanations for Semantics Extensions

A.3.1 Characterization of Maximal Explanations

Lemma. 6 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(a,b) \in \mathscr{R} \mid a, b \in S\}$. If $Expl_{Coh}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Coh, then $X \subseteq \mathscr{R}'$.*

Proof. (for Lemma 6) Suppose that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Coh*. Two cases must be considered.

Assume firstly that $X = \varnothing$. Then we trivially have $X \subseteq \mathscr{R}'$.

Assume secondly that $X \neq \varnothing$ and that $X \not\subseteq \mathscr{R}'$. Then, there exists $(a,b) \in \mathscr{R}$ such that $a, b \in S$ and $(a,b) \notin \mathscr{R}'$. Consider the graph $(\mathscr{A}'', \mathscr{R}'')$ with $\mathscr{A}'' = \mathscr{A}'$ and $\mathscr{R}'' = \mathscr{R}' \cup \{(a,b)\}$. Obviously, $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Coh*, by Def. 34, $\mathscr{A}' = S$, $\mathscr{R}' \subseteq X$ and because $X \neq \varnothing$, $\mathscr{R}' \neq \varnothing$ (Condition 3 in Def. 34). Thus, we deduce that $\mathscr{A}'' = S$, $\mathscr{R}'' \neq \varnothing$ and since $(a,b) \in X$, $\mathscr{R}'' \subseteq X$. Hence, by Def. 34, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Coh*, a contradiction with the supposition that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Coh*. □

Lemma. 7 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. If $Expl_{Def}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Def, then $Y \subseteq \mathscr{R}'$.*

Proof. (for Lemma 7) Suppose that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Def* and let $X = \{(b,a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$. Two cases must be considered.

Assume firstly that for all $b \in \mathscr{R}^{-1}(S)$, $b \notin \mathscr{R}^{+1}(S)$. In other words, $\nexists (a,b) \in \mathscr{R}$ such that $a \in S, b \in \mathscr{R}^{-1}(S)$. So, we trivially have $Y = \varnothing \subseteq \mathscr{R}'$.

Assume secondly that for some $b \in \mathscr{R}^{-1}(S)$, $b \in \mathscr{R}^{+1}(S)$. In this case, $Y \neq \varnothing$. Assume additionally that $Y \not\subseteq \mathscr{R}'$. Then, there exists $(x,y) \in \mathscr{R}$ such that $x \in S$, $y \in \mathscr{R}^{-1}(S)$ and $(x,y) \notin \mathscr{R}'$. Consider the graph $(\mathscr{A}'', \mathscr{R}'')$ with $\mathscr{A}'' = \mathscr{A}'$ and $\mathscr{R}'' = \mathscr{R}' \cup \{(x,y)\}$. Obviously, $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Def*, by Def. 36, $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$, $X \subseteq \mathscr{R}' \subseteq X \cup Y$ and for all $b \in \mathscr{R}^{-1}(S)$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}'$ with $a \in S$. Thus, we deduce that $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S)$ and since $(x,y) \in Y$, $X \subseteq \mathscr{R}'' \subseteq X \cup Y$. Moreover, it is obvious that for all $b \in \mathscr{R}^{-1}(S)$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}''$ with $a \in S$ (since $\mathscr{R}' \subseteq \mathscr{R}''$). Hence, by Def. 36, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Def*, a contradiction with the supposition that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Def*. □

Lemma. 8 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. If $Expl_{Rein1}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Rein1, then $X \subseteq \mathscr{A}'$.*

Proof. (for Lemma 8) Suppose that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein1*. Two cases must be considered.

Assume firstly that $(\mathscr{A} \setminus S) \cap X = \varnothing$. In other words, $S \cap X = X$. In this case, as $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*, by Def. 38, we have $S \cap X \subseteq \mathscr{A}'$ and so $X \subseteq \mathscr{A}'$.

Assume secondly that $(\mathscr{A} \setminus S) \cap X \neq \varnothing$ and that $X \not\subseteq \mathscr{A}'$. Then, there exists $x \in \mathscr{A}$ such that $\mathscr{R}^{-1}(x) = \varnothing$ (so $x \in X$), and $x \notin \mathscr{A}'$. Consider the graph $(\mathscr{A}'', \mathscr{R}'')$ with $\mathscr{A}'' = \mathscr{A}' \cup \{x\}$ and $\mathscr{R}'' = \mathscr{R}'$. Obviously, $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*, by Def. 38, $S \cap X \subseteq \mathscr{A}' \subseteq X$, and $\mathscr{R}' = \varnothing$. Thus, since $x \in X$, conditions 1 and 2 of Def. 38 are also satisfied for $(\mathscr{A}'', \mathscr{R}'')$: $S \cap X \subseteq \mathscr{A}'' \subseteq X$ and $\mathscr{R}'' = \varnothing$. In addition, as by assumption $(\mathscr{A} \setminus S) \cap X \neq \varnothing$, by Condition 3 of Def. 38 we know that $\exists a \in (\mathscr{A} \setminus S) \cap X$ such that $a \in \mathscr{A}'$, and so $a \in \mathscr{A}''$; thus Condition 3 of Def. 38 is also satisfied for $(\mathscr{A}'', \mathscr{R}'')$. Hence, by Def. 38, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein1*, a contradiction with the supposition that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein1*. □

Lemma. 9 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $Y = \{(a,b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. If $Expl_{Rein2}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for Rein2, then $Y \subseteq \mathscr{R}'$.*

Proof. (for Lemma 9) Suppose that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein2* and let $X = \{(b,c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$. Two cases must be considered.

Assume firstly that for all $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $b \notin \mathscr{R}^{+1}(S)$. In other words, $\nexists (a,b) \in \mathscr{R}$ such that $a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. So, we trivially have $Y = \varnothing \subseteq \mathscr{R}'$.

Assume secondly that for some $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $b \in \mathscr{R}^{+1}(S)$. In this case, $Y \neq \varnothing$. Assume additionally that $Y \not\subseteq \mathscr{R}'$. Then, there exists $(x,y) \in \mathscr{R}$ such that $x \in S$, $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $(x,y) \notin \mathscr{R}'$. Consider the graph $(\mathscr{A}'', \mathscr{R}'')$ with $\mathscr{A}'' = \mathscr{A}'$ and $\mathscr{R}'' = \mathscr{R}' \cup \{(x,y)\}$. Obviously, $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein2*, by Def. 39, $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$, $X \subseteq \mathscr{R}' \subseteq X \cup Y$ and for all $b \in \mathscr{R}^{-1}(S)$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}'$ with $a \in S$. Thus, we deduce that $\mathscr{A}'' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and since $(x,y) \in Y$, $X \subseteq \mathscr{R}'' \subseteq X \cup Y$. Moreover, it is obvious that for all $b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}''$ with $a \in S$ (since $\mathscr{R}' \subseteq \mathscr{R}''$). Hence, by Def. 39, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein2*, a contradiction with the supposition that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *Rein2*. □

Lemma. 10 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and consider $X = \{(a,b) \in \mathscr{R} \mid a \in S, b \notin S\}$. If $Expl_{CA}(S) = (\mathscr{A}', \mathscr{R}')$ is a maximal explanation for CA, then $X \subseteq \mathscr{R}'$.*

Proof. (for Lemma 10) Suppose that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *CA*. Two cases must be considered.

Assume firstly that for all $b \notin S$, $b \notin \mathscr{R}^{+1}(S)$. In other words, $\nexists (a,b) \in \mathscr{R}$ such that $a \in S, b \notin S$. So, we trivially have $X = \varnothing \subseteq \mathscr{R}'$.

Assume secondly that for some $b \notin S$, $b \in \mathscr{R}^{+1}(S)$. In this case, $X \neq \varnothing$. Assume additionally that $X \not\subseteq \mathscr{R}'$. Then, there exists $(x,y) \in \mathscr{R}$ such that $x \in S$, $y \notin S$ and $(x,y) \notin \mathscr{R}'$. Consider the graph $(\mathscr{A}'', \mathscr{R}'')$ with $\mathscr{A}'' = \mathscr{A}'$ and $\mathscr{R}'' = \mathscr{R}' \cup \{(x,y)\}$. Obviously, $(\mathscr{A}', \mathscr{R}')$ is a strict subgraph of $(\mathscr{A}'', \mathscr{R}'')$. Since $(\mathscr{A}', \mathscr{R}')$ is an explanation for *CA*, by Def. 41, $\mathscr{A}' = \mathscr{A}$, $\mathscr{R}' \subseteq X$ and for all $b \notin S$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}'$ with $a \in S$. Thus, we deduce that $\mathscr{A}'' = \mathscr{A}$ and since $(x,y) \in X$, $\mathscr{R}'' \subseteq X$. Moreover, it is obvious that for all $b \notin S$ such that $b \in \mathscr{R}^{+1}(S)$, $\exists (a,b) \in \mathscr{R}''$ with $a \in S$ (since $\mathscr{R}' \subseteq \mathscr{R}''$). Hence, by Def. 41, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *CA*, a contradiction with the supposition that $(\mathscr{A}', \mathscr{R}')$ is a maximal explanation for *CA*. □

A.3.2 Algorithms to Compute Minimal Explanations

Theorem. 10 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R})$ be an Argumentation Framework, $S \subseteq \mathscr{A}$ and $\pi \in \{$Coh, Def, Rein1, Rein2, CA$\}$. Algorithm Alg_π using $\mathcal{A}$ and S as inputs is sound and complete for the computation of a minimal explanation for π.*

Proof. (for Theorem 10)

• *Alg_{Coh}.* It begins by computing $(\mathscr{A}', \mathscr{R}') = MaxExpl_{Coh}(S)$, a maximal explanation for *Coh*. So, in particular, it is an explanation for *Coh*. Obviously $(x,y) \leftarrow choose(\mathscr{R}')$ implies that $(x,y) \in \mathscr{R}'$. This would mean that $\mathscr{R}' \setminus \{(x,y)\} \subset \mathscr{R}'$, and thus that $|\mathscr{R}' \setminus \{(x,y)\}| < |\mathscr{R}'|$. As such, lines 2-5 compute $(\mathscr{A}'', \mathscr{R}'')$ such that $\mathscr{A}'' = \mathscr{A}'$, $\mathscr{R}'' \subseteq \mathscr{R}'$ (in cases $\mathscr{R}' = \varnothing$ and $|\mathscr{R}'| = 1$, we have $\mathscr{R}'' = \mathscr{R}'$) and it holds that $|\mathscr{R}''| \leq 1$. Let $X = \{(a,b) \in \mathscr{R} \mid a, b \in S\}$. As $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Coh*, by Def. 34, we know that $\mathscr{A}' = S$ and $\mathscr{R}' \subseteq X$. But $\mathscr{A}'' = \mathscr{A}'$, so $\mathscr{A}'' = S$ and $\mathscr{R}'' \subseteq \mathscr{R}'$, so $\mathscr{R}'' \subseteq X$. In addition, if $\mathscr{R}' = \varnothing$, then $X = \varnothing$ by Def. 34, but $\mathscr{R}'' \subseteq X$, so we have $\mathscr{R}'' = \varnothing$ as well. Thus, by Def. 34, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Coh*. Moreover, we know that $|\mathscr{R}''| \leq 1$, so by Lem. 1, $(\mathscr{A}'', \mathscr{R}'')$ is a minimal explanation for *Coh*. So Alg_{Coh} is sound.

By Lem. 1, we know that $|\mathscr{R}''| \leq 1$. Alg_{Coh} begins by computing $(\mathscr{A}', \mathscr{R}') = MaxExpl_{Coh}(S)$, a maximal explanation for *Coh*. Since $(\mathscr{A}'', \mathscr{R}'')$ and $(\mathscr{A}', \mathscr{R}')$ are both explanations for *Coh*, by Def. 34, we know that $\mathscr{A}'' = S = \mathscr{A}'$. In addition, by Th. 9, we know that $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$. Let $X = \{(a,b) \in \mathscr{R} \mid a, b \in S\}$ and $(\mathscr{A}''', \mathscr{R}''')$ be the result computed by Alg_{Coh}. In the case where $|\mathscr{R}''| = 0$, by Def. 34 we have $X = \varnothing$, and thus, still by Def. 34, $\mathscr{R}' = \varnothing$. So, in this case, $\mathscr{R}' = \mathscr{R}''$. Lines 2-5 are ignored and Alg_{Coh} computes $(\mathscr{A}''', \mathscr{R}''')$ with $(\mathscr{A}''', \mathscr{R}''') = (\mathscr{A}', \mathscr{R}') = (\mathscr{A}'', \mathscr{R}'')$. In the case where $|\mathscr{R}''| = 1$, we denote $\mathscr{R}'' = \{(x_0, y_0)\}$. Since $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$, $\mathscr{R}'' \subseteq \mathscr{R}'$ and so $(x_0, y_0) \in \mathscr{R}'$. We denote $\mathscr{R}' = \{(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\}$. Obviously $(x,y) \leftarrow choose(\mathscr{R}')$ implies that $(x,y) \in \mathscr{R}'$. This would mean that $\mathscr{R}' \setminus \{(x,y)\} \subset \mathscr{R}'$, and thus that $|\mathscr{R}' \setminus \{(x,y)\}| < |\mathscr{R}'|$. As such, lines 2-5 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$ and

$\mathscr{R}''' = \mathscr{R}' \setminus \Delta$ with $|\mathscr{R}'''| = 1$. As we already know that $\mathscr{A}''' = \mathscr{A}''$, we only need to find a set Δ such that $\mathscr{R}' \setminus \Delta = \mathscr{R}''$. $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ is such a set. So Alg_{Coh} is complete.

• Alg_{Def}. It begins by computing $(A', R') = MaxExpl_{Def}(S)$, a maximal explanation for *Def*. So, in particular, it is an explanation for *Def*. Obviously $x \leftarrow choose(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$, for some $y \in \mathscr{R}^{-1}(S) \setminus S$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ (in cases $\mathscr{R}'^{-1}(y) = \varnothing$ and $|\mathscr{R}'^{-1}(y)| = 1$, we have $\mathscr{R}''^{-1}(y) = \mathscr{R}'^{-1}(y)$) and it holds that $|\mathscr{R}''^{-1}(y)| \leq 1$. Thus, lines 2-7 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$ and for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ and $|\mathscr{R}''^{-1}(y)| \leq 1$. Let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$. As $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Def*, by Def. 36, we know that $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$ and $X \subseteq \mathscr{R}' \subseteq X \cup Y$. But $\mathscr{A}'' = \mathscr{A}'$, so $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S)$, $\mathscr{R}'' \subseteq \mathscr{R}'$, so $\mathscr{R}'' \subseteq X \cup Y$, and as $y \in \mathscr{R}^{-1}(S) \setminus S$ and $(x, y) \in \mathscr{R}'$, we deduce that $(x, y) \in Y \setminus X$ and so $X \subseteq \mathscr{R}''$. In addition, if $\mathscr{R}'^{-1}(y) \cap S = \varnothing$, then $y \notin \mathscr{R}^{+1}(S)$ by Def. 36, but $\mathscr{R}'' \subseteq X \cup Y$, so we have $\mathscr{R}''^{-1}(y) \cap S = \varnothing$ as well. Thus, by Def. 36, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Def*. Moreover, we know that for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}''^{-1}(y)| \leq 1$, so by Lem. 2, $(\mathscr{A}'', \mathscr{R}'')$ is a minimal explanation for *Def*. So Alg_{Def} is sound.

By Lem. 2, we know that for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $|\mathscr{R}''^{-1}(y)| \leq 1$. Alg_{Def} begins by computing $(\mathscr{A}', \mathscr{R}') = MaxExpl_{Def}(S)$, a maximal explanation for *Def*. Since $(\mathscr{A}'', \mathscr{R}'')$ and $(\mathscr{A}', \mathscr{R}')$ are both explanations for *Def*, by Def. 36, we know that $\mathscr{A}'' = S \cup \mathscr{R}^{-1}(S) = \mathscr{A}'$. In addition, by Th. 9, we know that $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$. Let $X = \{(b, a) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(S), a \in S\}$, $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(S)\}$, $(\mathscr{A}''', \mathscr{R}''')$ be the result computed by Alg_{Def} and consider $y \in \mathscr{R}^{-1}(S) \setminus S$. In the case where $|\mathscr{R}''^{-1}(y)| = 0$, in particular, $\nexists (x, y) \in \mathscr{R}''$ with $x \in S$. So, by Def. 36 we have $y \notin \mathscr{R}^{+1}(S)$, and thus, still by Def. 36, $\mathscr{R}'^{-1}(y) = \varnothing$. So, in this case, $\mathscr{R}'^{-1}(y) = \mathscr{R}''^{-1}(y)$ and lines 3-6 are ignored. Thus, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$, $\mathscr{R}''' = \mathscr{R}'$ and so, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. In the case where $|\mathscr{R}''^{-1}(y)| = 1$, we denote $\mathscr{R}''^{-1}(y) = \{x_0\}$. Since $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$, $\mathscr{R}'' \subseteq \mathscr{R}'$, so $(x_0, y) \in \mathscr{R}'$ and in particular, $x_0 \in \mathscr{R}'^{-1}(y)$. We denote $\mathscr{R}'^{-1}(y) = \{x_0, x_1, \ldots, x_n\}$. Obviously $x \leftarrow choose(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}''' = \mathscr{R}' \setminus \Delta$ with $|\mathscr{R}'''^{-1}(y)| = 1$. So, we only need to find a set Δ such that $(\mathscr{R}' \setminus \Delta)^{-1}(y) = \mathscr{R}''^{-1}(y)$. $\{(x_1, y), \ldots, (x_n, y)\}$ is such a set. So, using $\Delta = \{(x_1, y), \ldots, (x_n, y)\}$ in the second case, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for some $y \in \mathscr{R}^{-1}(S) \setminus S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. Thus, lines 2-7 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. Since $X \subseteq \mathscr{R}''' \subseteq X \cup Y$ and for all $y \in \mathscr{R}^{-1}(S) \setminus S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$, we deduce that $\mathscr{R}''' = \mathscr{R}''$ and so $(\mathscr{A}''', \mathscr{R}''') = (\mathscr{A}'', \mathscr{R}'')$. So Alg_{Def} is complete.

• Alg_{Rein1}. It begins by computing $(\mathscr{A}', \mathscr{R}') = MaxExpl_{Rein1}(S)$, a maximal explanation for *Rein1*. So, in particular, it is an explanation for *Rein1*. Obviously $x \leftarrow choose(\mathscr{A}' \setminus S)$ implies that $x \in \mathscr{A}' \setminus S$. This would mean that $\mathscr{A}' \setminus \{x\} \subset \mathscr{A}'$, and thus that $|\mathscr{A}' \setminus \{x\}| < |\mathscr{A}'|$. In addition, since $x \in \mathscr{A}' \setminus S$, $|(\mathscr{A}' \setminus S) \setminus \{x\}| < |\mathscr{A}' \setminus S|$. As such, lines 2-5 compute $(\mathscr{A}'', \mathscr{R}'')$ such that $\mathscr{R}'' = \mathscr{R}'$, $\mathscr{A}'' \subseteq \mathscr{A}'$ (in cases $\mathscr{A}' \setminus S = \varnothing$ and $|\mathscr{A}' \setminus S| = 1$, we have $\mathscr{A}'' = \mathscr{A}'$) and it holds that $|\mathscr{A}'' \setminus S| \leq 1$. Let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$. As $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein1*, by Def. 38, we know that $\mathscr{R}' = \varnothing$ and $S \cap X \subseteq \mathscr{A}' \subseteq X$. But $\mathscr{R}'' = \mathscr{R}'$, so $\mathscr{R}'' = \varnothing$, $\mathscr{A}'' \subseteq \mathscr{A}'$, so $\mathscr{A}'' \subseteq X$ and since $x \in \mathscr{A}' \setminus S$, $S \cap X \subseteq \mathscr{A}''$. In addition, if $(\mathscr{A}' \setminus S) \cap X = \varnothing$, then $(\mathscr{A} \setminus S) \cap X = \varnothing$ by Def. 38, but $\mathscr{A}'' \subseteq X$, so we have $(\mathscr{A}'' \setminus S) \cap X = \varnothing$ as well. Thus, by Def. 38, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein1*. Moreover, we know that $|\mathscr{A}'' \setminus S| \leq 1$, so by Lem. 3, $(\mathscr{A}'', \mathscr{R}'')$ is a minimal explanation for *Rein1*. So Alg_{Rein1} is sound.

By Lem. 3, we know that $|\mathscr{A}'' \setminus S| \leq 1$. Alg_{Rein1} begins by computing $(\mathscr{A}', \mathscr{R}') = MaxExpl_{Rein1}(S)$, a maximal explanation for *Rein1*. Since $(\mathscr{A}'', \mathscr{R}'')$ and $(\mathscr{A}', \mathscr{R}')$ are both explanations for *Rein1*, by Def. 38, we know that $\mathscr{R}'' = \varnothing = \mathscr{R}'$. In addition, by Th. 9, we know that $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$. Let $X = \{a \in \mathscr{A} \mid \mathscr{R}^{-1}(a) = \varnothing\}$ and $(\mathscr{A}''', \mathscr{R}''')$ be the result computed by Alg_{Rein1}. In the case where $|\mathscr{A}'' \setminus S| = 0$, by Def. 38 we have $(\mathscr{A} \setminus S) \cap X = \varnothing$, and thus, $\mathscr{A}' \setminus S = \varnothing$. So, in this case, $\mathscr{A}' = S \cap X = \mathscr{A}''$. In addition, lines 2-5 are ignored and Alg_{Rein1} computes $(\mathscr{A}''', \mathscr{R}''')$ with $(\mathscr{A}''', \mathscr{R}''') = (\mathscr{A}', \mathscr{R}') = (\mathscr{A}'', \mathscr{R}'')$. In the case where $|\mathscr{A}'' \setminus S| = 1$, we denote $\mathscr{A}'' \setminus S = \{x_0\}$. Since $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$, $\mathscr{A}'' \subseteq \mathscr{A}'$ and so $x_0 \in \mathscr{A}'$. We

denote $\mathscr{A}' \setminus S = \{x_0, x_1, \ldots, x_n\}$. Obviously $x \leftarrow \mathit{choose}(\mathscr{A}' \setminus S)$ implies that $x \in \mathscr{A}' \setminus S$. This would mean that $\mathscr{A}' \setminus \{x\} \subset \mathscr{A}'$, and thus that $|\mathscr{A}' \setminus \{x\}| < |\mathscr{A}'|$. In addition, since $x \in \mathscr{A}' \setminus S$, $|(\mathscr{A}' \setminus S) \setminus \{x\}| < |\mathscr{A}' \setminus S|$. Thus, lines 2-5 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{R}''' = \mathscr{R}' = \mathscr{R}''$ and $\mathscr{A}''' \setminus S = (\mathscr{A}' \setminus S) \setminus \Delta$ with $|\mathscr{A}''' \setminus S| = 1$. As we already know that $\mathscr{R}''' = \mathscr{R}''$, we only need to find a set Δ such that $(\mathscr{A}' \setminus S) \setminus \Delta = \mathscr{A}'' \setminus S$. $\{x_1, \ldots, x_n\}$ is such a set. So $\mathit{Alg}_{\mathit{Rein1}}$ is complete.

• $\mathit{Alg}_{\mathit{Rein2}}$. It begins by computing $(\mathscr{A}', \mathscr{R}') = \mathit{MaxExpl}_{\mathit{Rein2}}(S)$, a maximal explanation for *Rein2*. So, in particular, it is an explanation for *Rein2*. Obviously $x \leftarrow \mathit{choose}(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$, for some $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ (in cases $\mathscr{R}'^{-1}(y) = \varnothing$ and $|\mathscr{R}'^{-1}(y)| = 1$, we have $\mathscr{R}''^{-1}(y) = \mathscr{R}'^{-1}(y)$) and it holds that $|\mathscr{R}''^{-1}(y)| \leq 1$. Thus, lines 2-7 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$ and for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ and $|\mathscr{R}''^{-1}(y)| \leq 1$. Let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$ and $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$. As $(\mathscr{A}', \mathscr{R}')$ is an explanation for *Rein2*, by Def. 39, we know that $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $X \subseteq \mathscr{R}' \subseteq X \cup Y$. But $\mathscr{A}'' = \mathscr{A}'$, so $\mathscr{A}'' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $\mathscr{R}'' \subseteq \mathscr{R}'$, so $\mathscr{R}'' \subseteq X \cup Y$, and as $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$ and $x \in S$, we deduce that $(x, y) \in Y \setminus X$ and so $X \subseteq \mathscr{R}''$. In addition, if $\mathscr{R}'^{-1}(y) = \varnothing$, then $y \notin \mathscr{R}^{+1}(S)$ by Def. 39, but $\mathscr{R}'' \subseteq X \cup Y$, so we have $\mathscr{R}''^{-1}(y) = \varnothing$ as well. Thus, by Def. 39, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *Rein2*. Moreover, we know that for all $y \in \mathscr{R}^{-1}(S) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}''^{-1}(y)| \leq 1$, so by Lem. 4, $(\mathscr{A}'', \mathscr{R}'')$ is a minimal explanation for *Rein2*. So $\mathit{Alg}_{\mathit{Rein2}}$ is sound.

By Lem. 4, we know that for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $|\mathscr{R}''^{-1}(y)| \leq 1$. $\mathit{Alg}_{\mathit{Rein2}}$, begins by computing $(\mathscr{A}', \mathscr{R}') = \mathit{MaxExpl}_{\mathit{Rein2}}(S)$, a maximal explanation for *Rein2*. Since $(\mathscr{A}'', \mathscr{R}'')$ and $(\mathscr{A}', \mathscr{R}')$ are both explanations for *Rein2*, by Def. 39, we know that $\mathscr{A}'' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) = \mathscr{A}'$. In addition, by Th. 9, we know that $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$. Let $X = \{(b, c) \in \mathscr{R} \mid b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), c \in \mathscr{R}^{+2}(S)\}$, $Y = \{(a, b) \in \mathscr{R} \mid a \in S, b \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$, $(\mathscr{A}''', \mathscr{R}''')$ be the result computed by $\mathit{Alg}_{\mathit{Rein2}}$ and consider $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$. In the case where $|\mathscr{R}''^{-1}(y)| = 0$, $\nexists (x, y) \in \mathscr{R}''$ with $x \in S$. So, by Def. 39 we have $y \notin \mathscr{R}^{+1}(S)$, and thus, still by Def. 39, $\mathscr{R}'^{-1}(y) = \varnothing$. So, in this case, $\mathscr{R}'^{-1}(y) = \mathscr{R}''^{-1}(y)$ and lines 3-6 are ignored. Thus, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$, $\mathscr{R}''' = \mathscr{R}'$ and so, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. In the case where $|\mathscr{R}''^{-1}(y)| = 1$, we denote $\mathscr{R}''^{-1}(y) = \{x_0\}$. Since $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$, $\mathscr{R}'' \subseteq \mathscr{R}'$, so $x_0 \in \mathscr{R}'$ and in particular, $x_0 \in \mathscr{R}'^{-1}(y)$. We denote $\mathscr{R}'^{-1}(y) = \{x_0, x_1, \ldots, x_n\}$. Obviously $x \leftarrow \mathit{choose}(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}''' = \mathscr{R}' \setminus \Delta$ with $|\mathscr{R}'''^{-1}(y)| = 1$. So, we only need to find a set Δ such that $(\mathscr{R}' \setminus \Delta)^{-1}(y) = \mathscr{R}''^{-1}(y)$. $\{(x_1, y), \ldots, (x_n, y)\}$ is such a set. So, using $\Delta = \{(x_1, y), \ldots, (x_n, y)\}$ in the second case, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for some $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. As such, lines 2-7 thus compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. Since $X \subseteq \mathscr{R}''' \subseteq X \cup Y$ and for all $y \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \setminus \mathscr{R}^{+2}(S)$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$, we deduce that $\mathscr{R}''' = \mathscr{R}''$ and so $(\mathscr{A}''', \mathscr{R}''') = (\mathscr{A}'', \mathscr{R}'')$. So $\mathit{Alg}_{\mathit{Rein2}}$ is complete.

• $\mathit{Alg}_{\mathit{CA}}$. It begins by computing $(\mathscr{A}', \mathscr{R}') = \mathit{Expl}_{\mathit{CA}}(S)$, a maximal explanation for *CA*. So, in particular, it is an explanation for *CA*. Obviously $x \leftarrow \mathit{choose}(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$, for some $y \in \mathscr{A} \setminus S$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ (in cases $\mathscr{R}'^{-1}(y) = \varnothing$ and $|\mathscr{R}'^{-1}(y)| = 1$, we have $\mathscr{R}''^{-1}(y) = \mathscr{R}'^{-1}(y)$) and it holds that $|\mathscr{R}''^{-1}(y)| \leq 1$. Thus, lines 2-7 compute $(\mathscr{A}'', \mathscr{R}'')$ such that, $\mathscr{A}'' = \mathscr{A}'$ and for all $y \notin S$, $\mathscr{R}''^{-1}(y) \subseteq \mathscr{R}'^{-1}(y)$ and $|\mathscr{R}''^{-1}(y)| \leq 1$. Let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$. As $(\mathscr{A}', \mathscr{R}')$ is an explanation for *CA*, by Def. 41, we know that $\mathscr{A}' = \mathscr{A}$ and $\mathscr{R}' \subseteq X$. But $\mathscr{A}'' = \mathscr{A}'$, so $\mathscr{A}'' = \mathscr{A}$ and $\mathscr{R}'' \subseteq \mathscr{R}'$, so $\mathscr{R}'' \subseteq X$. In addition, if $\mathscr{R}'^{-1}(y) \cap S = \varnothing$, then $y \notin \mathscr{R}^{+1}(S)$ by Def. 41, but $\mathscr{R}'' \subseteq X$, so we have $\mathscr{R}''^{-1}(y) \cap S = \varnothing$ as well. Thus, by Def. 41, $(\mathscr{A}'', \mathscr{R}'')$ is an explanation for *CA*. Moreover, we know that for all $y \notin S$, $|\mathscr{R}''^{-1}(y)| \leq 1$, so by Lem. 5, $(\mathscr{A}'', \mathscr{R}'')$ is a minimal explanation for *CA*. So $\mathit{Alg}_{\mathit{CA}}$ is sound.

By Lem. 5, we know that for all $y \notin S$, $|\mathscr{R}''^{-1}(y)| \leq 1$. Alg_{CA} begins by computing $(\mathscr{A}', \mathscr{R}') = Expl_{CA}(S)$, a maximal explanation for CA. Since $(\mathscr{A}'', \mathscr{R}'')$ and $(\mathscr{A}', \mathscr{R}')$ are both explanations for CA, by Def. 41, we know that $\mathscr{A}'' = \mathscr{A} = \mathscr{A}'$. In addition, by Th. 9, we know that $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$. Let $X = \{(a, b) \in \mathscr{R} \mid a \in S, b \notin S\}$, $(\mathscr{A}''', \mathscr{R}''')$ be the result computed by Alg_{CA} and consider $y \notin S$. In the case where $|\mathscr{R}''^{-1}(y)| = 0$, in particular, $\nexists (x, y) \in \mathscr{R}''$ with $x \in S$. So, by Def. 41 we have $y \notin \mathscr{R}^{+1}(S)$, and thus, still by Def. 41, $\mathscr{R}'^{-1}(y) = \varnothing$. So, in this case, $\mathscr{R}'^{-1}(y) = \mathscr{R}''^{-1}(y)$ and lines 3-6 are ignored. Thus, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$, $\mathscr{R}''' = \mathscr{R}'$ and so, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. In the case where $|\mathscr{R}''^{-1}(y)| = 1$, we denote $\mathscr{R}''^{-1}(y) = \{x_0\}$. Since $(\mathscr{A}'', \mathscr{R}'') \subseteq (\mathscr{A}', \mathscr{R}')$, $\mathscr{R}'' \subseteq \mathscr{R}'$, so $x_0 \in \mathscr{R}'$ and in particular, $x_0 \in \mathscr{R}'^{-1}(y)$. We denote $\mathscr{R}'^{-1}(y) = \{x_0, x_1, \ldots, x_n\}$. Obviously $x \leftarrow choose(\mathscr{R}'^{-1}(y))$ implies that $x \in \mathscr{R}'^{-1}(y)$. In particular, it implies that $(x, y) \in \mathscr{R}'$. This would mean that $\mathscr{R}'^{-1}(y) \setminus \{(x, y)\} \subset \mathscr{R}'^{-1}(y)$, and thus that $|\mathscr{R}'^{-1}(y) \setminus \{(x, y)\}| < |\mathscr{R}'^{-1}(y)|$. As such, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}' = \mathscr{A}''$ and $\mathscr{R}''' = \mathscr{R}' \setminus \Delta$ with $|\mathscr{R}'''^{-1}(y)| = 1$. So, we only need to find a set Δ such that $(\mathscr{R}' \setminus \Delta)^{-1}(y) = \mathscr{R}''^{-1}(y)$. $\{(x_1, y), \ldots, (x_n, y)\}$ is such a set. So, using $\Delta = \{(x_1, y), \ldots, (x_n, y)\}$ in the second case, lines 3-6 compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for some $y \notin S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. As such, lines 2-7 thus compute $(\mathscr{A}''', \mathscr{R}''')$ such that $\mathscr{A}''' = \mathscr{A}''$ and for all $y \notin S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$. Since $\mathscr{R}''' \subseteq X$ and for all $y \notin S$, $\mathscr{R}'''^{-1}(y) = \mathscr{R}''^{-1}(y)$, we deduce that $\mathscr{R}''' = \mathscr{R}''$ and so $(\mathscr{A}''', \mathscr{R}''') = (\mathscr{A}'', \mathscr{R}'')$. So Alg_{CA} is complete. □

Appendix B

Proofs of Chapter 4

B.1 Conventions

We recall here the conventions used, especially when handling logical objects. Please note that, when doing so, we handle mathematical objects at three different levels:

- The elements of the argumentation framework at hand (that is, elements of $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$)
- The elements of the logical language
- The elements of the domain of a first-order interpretation

As explained in Section 4.5.1, we take the letter denoting an element of $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ to be also used as the individual constant representing this element in the logical language. However, we make a distinction with the elements of the domain, and will thus use a dotted notation $\dot{e}$ to indicate the element of the domain that is mapped to the individual constant e (and thus to the element e of $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$).

When working with the logical language, letters x, y, z, u, v denote individual variables, while letters $e, e_1, e_2, e_3, e_4, e_5$ denote individual constants as well as elements of $\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$.

B.2 Proofs for Section 4.5.1: A Generic Theory

Theorem. 11 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C. Let φ be a formula of $\mathfrak{L}_{\mathcal{A}}$. $\mathcal{S}_{\mathcal{A}}, UDC_{\mathcal{A}} \models \varphi$ if and only if $\varphi \in \mathrm{Th}(\mathfrak{H}(\mathcal{S}_{\mathcal{A}}))$.*

Proof. (for Theorem 11)

$\Rightarrow$ The "only if" direction is obvious.

$\Leftarrow$ For the "if" direction, by an *absurdio ad reductio*, we assume that $\varphi \in \mathrm{Th}(\mathfrak{H}(\mathcal{S}_{\mathcal{A}}))$ and $\mathcal{S}_{\mathcal{A}}, UDC_{\mathcal{A}} \not\models \varphi$ for a given φ.

Thus, there exists a model $M = (D_M, \sigma_M)$ of $\mathcal{S}_{\mathcal{A}}$ and $UDC_{\mathcal{A}}$ such that $M \models \neg\varphi$. Following $UDC_{\mathcal{A}}$ and Axiom 4.1d, $|D_M| = |\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}|$. Moreover, following Axiom 4.1d we can consider the restriction of σ_M over constants, so $\sigma_M|_{\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}} : (\mathscr{A} \cup \mathscr{R} \cup \mathscr{S}) \to D_M$ is injective, and so, bijective. Thus D_M can be viewed as an enumeration such that $D_M = \{d_1, \ldots, d_l\}$ with $\sigma_M(e_i) = d_i$. Since the restriction of σ_M over constants is a bijection to the domain of M, we can use it as a naming function $\bar{\cdot}$ (i.e., e_i is used in place of $\overline{d_i}$ and the language of the model M is considered as being only $\mathfrak{L}_{\mathcal{A}}$).

A Herbrand model H for $\mathfrak{L}_{\mathcal{A}}$ can be built as follows:
First, the domain of H is $\{e_1, \ldots, e_{n_1}, e_{n_1+1}, \ldots e_{n_2}, e_{n_2+1}, \ldots e_{n_3}\}$. Since H is a Herbrand model, $\sigma_H(e_i) = e_i$.

Let prove by induction that for any formula φ, $H \models \varphi$ iff $M \models \varphi$.

First consider atomic formulas. Two cases are possible:

- Let us show that $H \models (t_1 = t_2)$ iff $M \models (t_1 = t_2)$. Since $\mathfrak{L}_{\mathcal{A}}$ is the language of M and the language of H, t_1 and t_2 can only be e_i. So, $(t_1 = t_2)$ amounts to $(e_i = e_j)$ for some i and j. Then, $H \models (e_i = e_j)$ iff $\sigma_H(e_i) = \sigma_H(e_j)$ iff $e_i = e_j$ iff $i = j$ iff $d_i = d_j$ iff $\sigma_M(e_i) = \sigma_M(e_j)$ iff $M \models (e_i = e_j)$.
- Consider $\sigma_H(P)$ such that $\sigma_H(P)(e_i, \ldots, e_j) = \sigma_M(P)(d_i, \ldots, d_j)$. Let us show that $H \models P(t_1, \ldots, t_p)$ iff $M \models P(t_1, \ldots, t_p)$. Once again, each t_i can only be a certain e_l. So, $P(t_1, t_2, \ldots, t_p)$ amounts to $P(e_i, e_j, \ldots, e_k)$ for some i, j, $\ldots$, and k.
  Then, $H \models P(e_i, e_j, \ldots, e_k)$ iff $\sigma_H(P)(e_i, e_j, \ldots, e_k) = \top$ iff $\sigma_M(P)(d_i, d_j, \ldots, d_k) = \top$ iff $\sigma_M(P)(\sigma_M(e_i), \sigma_M(e_j), \ldots, \sigma_M(e_k)) = \top$ iff $M \models P(e_i, e_j, \ldots, e_k)$.

So, for any atomic formula A of $\mathfrak{L}_{\mathcal{A}}$, we have $H \models A$ iff $M \models A$.

Secondly, consider more complex formulas (i.e., formulas that are not atomic). The case of the connectives $\neg$ and $\vee$ is obvious. More interesting is the case of the quantifiers. On the one hand, $H \models \forall x\ B(x)$ iff $H \models B(\overline{x})$ for any $x \in \mathscr{A} \cup \mathscr{R} \cup \mathscr{S}$ iff $H \models B(x_i)$ for $i = 1, \ldots, n$. On the other hand, $M \models \forall x\ B(x)$ iff $M \models B(\overline{d})$ for any $d \in D_M$ iff $M \models B(e_i)$ for $i = 1, \ldots, n_3$. Assume that, by the induction assumption, $H \models B(e_i)$ iff $M \models B(e_i)$. So, $H \models \forall x\ B(x)$ iff $M \models \forall x\ B(x)$.

Hence, M being a model of $\mathcal{S}_{\mathcal{A}}$, this implies that H is also a model of $\mathcal{S}_{\mathcal{A}}$. Moreover, considering $\varphi \in \mathrm{Th}(\mathfrak{H}(\mathcal{S}_{\mathcal{A}}))$, then $H \models \varphi$ and thus $M \models \varphi$, a contradiction arises.

□

B.3 Proofs for Section 4.5.2: Simplification and specialisations

Lemma 13. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ that satisfies $\forall x\ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$.*

Proof. (for Lemma 13) Since I is a Herbrand model of (4.1) and because $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$, I is a model of $\forall x([Arg(x) \vee Att(x)] \rightarrow PrimaFacie(x))$. Also, I is a model of $\forall x\ (Cand(x) \rightarrow Arg(x) \vee Att(x))$, hence I is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$. □

Proposition. 6 *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ that satisfies $\forall x\ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.6) if and only if it is a model of $\forall x \in Cand\ (Supported(x))$.*

Proof. (for Proposition 6) We prove both directions of the equivalence.

⇒ Let I be a Herbrand model of, (4.1), (4.2), (4.3), of $\forall x\ \big(Cand(x) \rightarrow Arg(x) \vee Att(x)\big)$, and of (4.6). By lemma 13, I is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$. That is, $\forall x \in Cand\ (PrimaFacie(x))$. However, I is a Herbrand model of (4.6), i.e., $\forall x \in Cand\ (\ [PrimaFacie(x) \vee \exists \alpha \in Sup(T(\alpha, x) \rightarrow Activable(\alpha))] \rightarrow Supported(x)\ )$.
Thus, we can conclude that I is a model of $\forall x \in Cand\ \ (Supported(x))$.

⇐ Let I be a Herbrand model of (4.1), (4.2), (4.3), of $\forall x\ \big(Cand(x) \rightarrow Arg(x) \vee Att(x)\big)$, and of $\forall x \in Cand\ (Supported(x))$. Therefore, (4.6) is true for I, i.e., I satisfies (4.6), as desired.

□

Proposition. 7 *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ and (4.6) that satisfies $\forall x\ (Cand(x) \rightarrow (Arg(x) \vee Att(x)))$ is a model of (4.7a), i.e., $\forall x \in Cand\ \big(Supported(x) \rightarrow [PrimaFacie(x) \vee \exists \alpha \in Sup\ (T(\alpha, x) \wedge Activable(\alpha))]\big)$*

Proof. (for Proposition 7) Let I be a Herbrand model of (4.1), (4.2), (4.3), (4.6) and a model of of $\forall x \ \big(Cand(x) \rightarrow Arg(x) \vee Att(x)\big)$. By Proposition 6, I is a model of $\forall x \in Cand \ (Supported(x))$. For I to be a model of (4.7a), we show that I is a model of $\forall x \in Cand(PrimaFacie(x) \vee \exists \alpha \in Sup(T(\alpha, x) \wedge Activable(\alpha)))$. By lemma 13, I is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$. Thus, I is a model of $\forall x \in Cand(PrimaFacie(x) \vee \exists \alpha \in Sup(T(\alpha, x) \wedge Activable(\alpha)))$, as desired. □

Proposition. 8 *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, each Herbrand model of $\Sigma(\mathcal{A})$ and (4.6) that satisfies $\forall x \ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.7b) if and only if it is a model of $\forall x \in Cand \ (\neg Unsupportable(x))$.*

Proof. (for Proposition 8) We prove both directions of the equivalence.

⇒ Let I be a Herbrand model of (4.1), (4.2), (4.3), (4.6), of $\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \vee Att(x)\big)$ and of (4.7b).
By lemma 13, I is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$. Moreover, I is a Herbrand model of (4.7b), i.e.:
$\forall x \in Cand(\ Unsupportable(x) \leftrightarrow [\neg PrimaFacie(x) \wedge \forall \alpha \in Sup(T(\alpha, x) \rightarrow Desactivated(\alpha))] \ )$.
Thus, we can conclude that I is a model of $\forall x \in Cand \ (\neg Unsupportable(x))$.

⇐ Let I be a Herbrand model of (4.1), (4.2), (4.3), (4.6), of $\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \vee Att(x)\big)$ and of $\forall x \in Cand \ (\neg Unsupportable(x))$. For I to be a model of (4.7a), we show that I is not a model of $\forall x \in Cand(\neg PrimaFacie(x) \wedge \forall \alpha \in Sup(T(\alpha, x) \rightarrow Desactivated(\alpha)))$. By lemma 13, I is a model of $\forall x(Cand(x) \rightarrow PrimaFacie(x))$. I is thus not a model of $\forall x(Cand(x) \rightarrow \neg PrimaFacie(x))$, and so, it is not a model of $\forall x \in Cand(\neg PrimaFacie(x) \wedge \forall \alpha \in Sup(T(\alpha, x) \rightarrow Desactivated(\alpha)))$, as desired.

□

Proposition. 9 *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C and suppose that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Then, every Herbrand model of $\Sigma(\mathcal{A})$ that satisfies $\forall x \ (Cand(x) \rightarrow Arg(x) \vee Att(x))$ is a model of (4.10b), i.e., $\forall x \in Cand \ \big(\neg Supported(x) \rightarrow Unsupportable(x)\big)$*

Proof. (for Proposition 9) Let I be a Herbrand model of (4.1), (4.2), (4.3) and a model of $\forall x \ \big(Cand(x) \leftrightarrow Arg(x) \vee Att(x)\big)$. By Proposition 6, I is a model of $\forall x \in Cand \ (Supported(x))$. Trivially, I is thus a model of (4.10b). □

B.4 Proofs for Section 4.5.3: Theory for AF

Proposition. 10 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an AF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

Proof. (for Proposition 10)

The idea is to apply a succession of operations on the language used in this work to retrieve the language that is used in the "Logical Description of a RAF" and "Case of AF" sections of [CL20]. In the following, when we mention formulas from [CL20], we refer to formulas given in these particular sections. These operations will then be applied successively on the formulas (4.4) and $(4.5)_{\text{AF}}$ to $(4.10)_{\text{AF}}$ in order to eventually yield the formulas used in [CL20].

First operation (denoted Θ): rename correctly some of the predicates of the language as follows.

- Arg becomes Arg
- Att becomes $Attack$
- Sup becomes $\bot$ (Sup has no corresponding predicate in [CL20] and in an AF, we have $\mathscr{S} = \varnothing$)
- $PrimaFacie$ becomes $\top$ ($PrimaFacie$ has no corresponding predicate in the logical encoding proposed by [CL20] for AF; moreover, in an AF, we consider that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$)
- $Acceptable$ becomes $Acceptable$ (we will see later that this predicate can be replaced by others)
- $Selected(x)$ becomes $Acc(x)$
- $Supported$ becomes $\top$ (Sup has no corresponding predicate in [CL20] and we can use proposition 6)
- $Unsupportable$ becomes $\bot$ (Sup has no corresponding predicate in [CL20] and we can use proposition 8)
- $Unacceptable$ becomes $NAcc$
- $Cand$ becomes $\top$ ($Cand$ does not appear in formulas (4.4) and $(4.5)_{\text{AF}}$ to $(4.10)_{\text{AF}}$ and has no corresponding predicate in [CL20])
- $Activable$ becomes $\top$ (same reason as for $Cand$))
- $Defeated$ becomes $\top$ (same reason as for $Cand$)
- $Inhibited$ becomes $\top$ (same reason as for $Cand$)
- $Desactivated$ becomes $\top$ (same reason as for $Cand$)

Second operation (denoted Π : replace the use of binary predicates S and T by the introduction of functional terms s_α and t_α (justified by the presence of the axioms (4.11) and (4.12) in the theory).

- $\forall a \in Arg(S(\alpha, a) \rightarrow \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $\exists a \in Arg(S(\alpha, a) \wedge \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $T(\alpha, x)$ becomes $t_\alpha = x$

The idea is then to apply successively Θ and Π on formulas (4.4) and $(4.5)_{\text{AF}}$ to $(4.10)_{\text{AF}}$ so that we obtain the formulas used in [CL20]. However, for some formulas, this result is not immediate. We will therefore use different versions of these formulas (namely $(4.5\text{a})_{\text{AF}}$) on which to apply Θ and Π.

Concerning Formula $(4.5\text{a})_{\text{AF}}$, we change the subformula $\exists x \in Arg\big(T(\alpha, x) \wedge Unacceptable(x)\big)$ into $\forall x \in Arg\big(T(\alpha, x) \rightarrow Unacceptable(x)\big)$. This modification is only valid because in propositions 1 to 6 we consider models of theories that contain (4.12). We then use standard modifications that preserve logical equivalence to put the quantifier $\forall x \in Arg$ at the beginning of the formula. It results in Formula $(4.5\text{a})_{\text{Diff}}$ which will be used instead of $(4.5\text{a})_{\text{AF}}$.

$$\forall x \in Arg\Big(\forall \alpha \in Att\Big(\Big[\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge T(\alpha, x)\Big] \rightarrow Unacceptable(x)\ \Big)\Big) \qquad ((4.5a)_{\text{Diff}})$$

By applying successively Θ and Π on formulas (4.4), $(4.5a)_{\text{Diff}}$, $(4.5b)_{\text{AF}}$, $(4.8)_{\text{AF}}$, $(4.9)_{\text{AF}}$, $(4.10)_{\text{AF}}$, we obtain the following formula.

$$\forall x(Acc(x) \leftrightarrow (Acceptable(x) \wedge \top)) \qquad ((4.4)_{\text{Shift}})$$

The result of this previous formula is that the predicate *Acceptable* becomes equivalent to the predicate *Acc*. In the following formulas we will replace the predicate *Acceptable* by the predicate *Acc*.

$$\forall x \in Arg(\forall \alpha \in Attack([Acc(s_\alpha) \wedge (t_\alpha = x)] \rightarrow NAcc(x))) \qquad ((4.5a)_{\text{ShiftA}})$$

$$\forall x \in Arg(NAcc(x) \rightarrow \neg Acc(x)) \qquad ((4.5b)_{\text{ShiftA}})$$

$$\forall x \in Arg(Acc(x) \rightarrow \forall \alpha \in Attack((t_\alpha = x) \rightarrow \exists \beta \in Attack((t_\beta = s_\alpha) \wedge Acc(s_\beta)))) \qquad ((4.8)_{\text{ShiftA}})$$

$$\forall x \in Arg(\forall \alpha \in Attack((t_\alpha = x) \rightarrow \exists \beta \in Attack((t_\beta = s_\alpha) \wedge Acc(s_\beta))) \rightarrow Acc(x)) \qquad ((4.9)_{\text{ShiftA}})$$

$$\forall x \in Arg(\neg Acc(x) \rightarrow \exists \alpha \in Attack(t_\alpha = x \wedge Acc(s_\alpha))) \qquad ((4.10a)_{\text{ShiftA}})$$

These modifications are not enough to retrieve the language used for AF in [CL20]. Indeed, the formulas used in [CL20] do not use universal and existential quantifiers, and instead use conjunctions and disjunctions that range over some set of arguments. We thus need a way to transform our quantifiers into propositional operators. This will be the role of a third operation (denoted Ω). The idea is to turn universal quantifiers into conjunctions and existential quantifiers into disjunctions. The justification for doing so is that we are interested in encoding finite argumentation frameworks and that in propositions 1 to 6 we consider Herbrand models of theories that contain (4.2).

In addition, some of our quantifiers range over attacks, while the conjunctions and disjunctions in the formulas of [CL20] only range over arguments. The idea here is to use the successor and predecessor functions $\mathscr{R}^+$ and $\mathscr{R}^-$. We can then define Ω.

Third operation (Ω) : replace first order quantifiers by propositional operators.

- $\forall x \in Arg\ (\varphi)$ becomes $\bigwedge_{x \in \mathscr{A}}(\varphi)$
- $\exists x \in Arg\ (\varphi)$ becomes $\bigvee_{x \in \mathscr{A}}(\varphi)$
- $\forall \alpha \in Attack\ (\varphi)$ such that $t_\alpha = x$ occurs in φ becomes $\bigwedge_{y \in \mathscr{R}^-(x)}(\varphi)$ with all occurrences of t_α in φ replaced by x and all occurrences of s_α in φ replaced by y

- $\exists \alpha \in \mathit{Attack}\ (\varphi)$ such that $t_\alpha = x$ occurs in φ becomes $\bigvee_{y \in \mathscr{R}^-(x)}(\varphi)$ with all occurrences of t_α in φ replaced by x and all occurrences of s_α in φ replaced by y

By applying Ω on formulas $(4.5)_{\text{ShiftA}}$, $(4.8)_{\text{ShiftA}}$, $(4.9)_{\text{ShiftA}}$, $(4.10\text{a})_{\text{ShiftA}}$, we obtain the following formulas.

$$\bigwedge_{x \in \mathscr{A}} (\bigwedge_{y \in \mathscr{R}^-(x)} ([Acc(y) \wedge (x = x)] \rightarrow NAcc(x))) \qquad ((4.5\text{a})_{\text{ShiftB}})$$

$$\bigwedge_{x \in \mathscr{A}} (NAcc(x) \rightarrow \neg Acc(x)) \qquad ((4.5\text{b})_{\text{ShiftB}})$$

$$\bigwedge_{x \in \mathscr{A}} (Acc(x) \rightarrow \bigwedge_{y \in \mathscr{R}^-(x)} ((x = x) \rightarrow \bigvee_{z \in \mathscr{R}^-(y)} ((y = y) \wedge Acc(z)))) \qquad ((4.8)_{\text{ShiftB}})$$

$$\bigwedge_{x \in \mathscr{A}} (\bigwedge_{y \in \mathscr{R}^-(x)} ((x = x) \rightarrow \bigvee_{z \in \mathscr{R}^-(y)} ((y = y) \wedge Acc(z))) \rightarrow Acc(x)) \qquad ((4.9)_{\text{ShiftB}})$$

$$\bigwedge_{x \in \mathscr{A}} (\neg Acc(x) \rightarrow \bigvee_{y \in \mathscr{R}^-(x)} ((x = x) \wedge Acc(y))) \qquad ((4.10\text{a})_{\text{ShiftB}})$$

We have the following immediate results (where "amounts" means "logically equivalent").

- Formula $(4.5\text{a})_{\text{ShiftB}}$ amounts to Formula (1) of [CL20]
- Formula $(4.5\text{b})_{\text{ShiftB}}$ amounts to Formula (3) of [CL20]
- Formula $(4.8)_{\text{ShiftB}}$ amounts to Formula (11) of [CL20]
- Formula $(4.9)_{\text{ShiftB}}$ amounts to Formula (13) of [CL20]
- Formula $(4.10\text{a})_{\text{ShiftB}}$ amounts to Formula (15) of [CL20]

The only missing formulas are those that are used to describe the graph of an argumentation framework, namely, (5), (7) and (9) in [CL20]. Since in [CL20], the only constant symbols used are those representing the arguments, formulas (4), (6) and (8) are ignored. Moreover, Formula (5) becomes $\forall x(Arg(x))$. They should be retrieved using formulas (4.1), (4.2) and (4.3).

Let us consider formulas $(4.1)_{\text{Shift}}$, $(4.2)_{\text{Shift}}$ and $(4.3)_{\text{Shift}}$, which are the formulas obtained by applying Θ on formulas (4.1), (4.2) and (4.3). We have the following results.

- Formulas $(4.1\text{a})_{\text{Shift}}$ and $(4.2\text{b})_{\text{Shift}}$ together amount to Formula (7) of [CL20]
- Formula $(4.1\text{d})_{\text{Shift}}$ amounts to formulas (9) and (10) together of [CL20] (although (10) is not used in [CL20], we still encompass (9))

The problem with Formula (5) lies in the fact that [CL20] uses only the arguments as constant symbols, while we use arguments and attacks as such, even in the case of an AF. This makes our theory to fail satisfying the formula $\forall x(Arg(x))$. However, we have seen with formulas $(4.5\text{a})_{\text{ShiftB}}$ to $(4.10\text{a})_{\text{ShiftB}}$ that, for a given semantics, we can build from our initial theory another one in which the constant symbols representing the

attacks are not used at all (i.e., there is not even a single occurrence). We could then delete these constant symbols (along with the formulas / subformulas and predicates (like S and T) that use them, and get the same results. In this new theory, only the constant symbols representing the arguments remain, and thus the axiom $\forall x(Arg(x))$ holds ((4.1), (4.2)).

In conclusion, since our theory is equivalent to the theory given in [CL20], Consequence 4.1 of [CL20] and Proposition 10 are also equivalent. And so Proposition 10 holds. □

B.5 Proofs for Section 4.5.4: Theory for AF-C

Proposition. 11 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an AF-C and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a $\subseteq$-maximal Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a $\subseteq$-minimal Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

Proof. (for Proposition 11)

1. For conflict-freeness:

   ⇒ Consider a set S of $\mathcal{A}$ that is *conflict-free*. An Herbrand interpretation I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ can be defined as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \top$ iff $e \in \mathscr{S}$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R} \cup \mathscr{S}$ and $e_2 \in s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R} \cup \mathscr{S}$ and $e_2 \in t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \top$ iff $e \in \mathscr{P}$
   - For any $\dot{e}$ in D_I, $I(Supported(e)) = \top$ and $I(Unsupportable(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Acceptable(e)) = \top$ iff $I(Selected(e)) = \top$
   - For any $\dot{e}$ in D_I, $I(Unacceptable(e)) = \top$ iff $I(Acceptable(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Cand(e)) = \top$ iff $I(Arg(e)) = \top$
   - For any $\dot{e_1}$ in D_I, $I(Activable(e_1)) = \top$ iff for any $\dot{e_2}$ in D_I such that $I(Arg(e_2)) = \top$, if $I(S(e_1, e_2)) = \top$ then $I(Selected(e_2)) = \top$
   - For any $\dot{e_1}$ in D_I, $I(Defeated(e_1)) = \top$ iff there exists $\dot{e_2}$ in D_I such that $I(Att(e_2)) = \top$, $I(T(e_2, e_1)) = \top$ and $I(Activable(e_2)) = \top$
   - For any $\dot{e_1}$ in D_I, $I(Inhibited(e_1)) = \top$ iff there exists $\dot{e_2}$ in D_I such that $I(Arg(e_2)) = \top$, $I(S(e_1, e_2)) = \top$ and $I(Defeated(e_2)) = \top$

– For any $\dot{e}$ in D_I, $I(Desactivated(e)) = \top$ iff $I(Inhibited(e)) = \top$

With this definition, $S_I = S$. It remains to prove that I is a model of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.12), (4.13)\}$. By definition of I and because $t : \mathscr{R} \cup \mathscr{S} \mapsto \mathscr{A}$ ($\mathcal{A}$ being an AF-C), I is a model of Axiom (4.12). Moreover I is a model of formulas (4.13).

Let prove that I is a model of $\Sigma_{Coh}(\mathcal{A})$. Obviously I is a model of formulas (4.1), (4.2) and (4.3). Since I is a model of formulas (4.13), we must prove that I is a model of formulas (4.4), $(4.5)_{\text{AF}}$ and (4.6). By definition of I, formulas (4.4), $(4.5\text{b})_{\text{AF}}$ and (4.6) are obviously satisfied.

Consider now Formula $(4.5\text{a})_{\text{AF}}$. Assume that I does not satisfy Formula $(4.5\text{a})_{\text{AF}}$. So there exists $\dot{e_2} \in D_I$ such that, $I(Att(e_2)) = \top$, for any $\dot{e_3} \in D_I$ such that $I(S(e_2, e_3)) = \top$, $I(Selected(e_3)) = \top$, and for any $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$ and $I(T(e_2, e_1)) = \top$, $I(Unacceptable(e_1)) = \bot$. By definition of I, this implies that, for any $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$ and $I(T(e_2, e_1)) = \top$, $I(Acceptable(e_1)) = \top$, and so that $I(Selected(e_1)) = \top$. Since I is a model of Axiom (4.12), $\dot{e_1}$ is unique. So there exists $e_2 \in \mathscr{R}$ such that $s(e_2) \subseteq S$ and $t(e_2) = e_1$ with $e_1 \in S$. That is in contradiction with S *conflict-free* and thus Formula $(4.5\text{a})_{\text{AF}}$ is satisfied by I.

$\Leftarrow$ Let I is a Herbrand model of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.12), (4.13)\}$. Assume that S_I is not *conflict-free*. So, there exists $e_2 \in \mathscr{R}$ and $e_1 \in \mathscr{A}$ such that $s(e_2) \subseteq S_I$, $e_1 \in S_I$, and $t(e_2) = e_1$. Following formulas (4.1) that are satisfied by I, there exist $\dot{e_2}, \dot{e_1} \in D_I$ such that $I(Att(e_2)) = \top$ and $I(Arg(e_1)) = \top$. Moreover, since I is a model of Formula (4.3), this implies that $I(T(e_2, e_1)) = \top$ and that for any $\dot{e_3} \in D_I$ such that $e_3 \in s(e_2)$, $I(S(e_2, e_3)) = \top$. And, by the definition of S_I, $I(Selected(e_1)) = \top$ and for any $\dot{e_3} \in D_I$ such that $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Selected(e_3)) = \top$. Since I is a model of Formula (4.4), $I(Acceptable(e_1)) = \top$, and so, following Formula $(4.5\text{b})_{\text{AF}}$ that is satisfied by I, $I(Unacceptable(e_1)) = \bot$. Thus, there exists $\dot{e_2} \in D_I$ such that $I(Att(e_2)) = \top$ and I is a model of Formula $\forall z \in Arg(S(e_2, z) \rightarrow Selected(z))$. Since I is a model of Axiom (4.12), I is also a model of the formula $\exists x \in Arg(\ T(e_2, x) \wedge \neg Unacceptable(x)\ )$, that is in contraction with Formula $(4.5\text{a})_{\text{AF}}$, and with the assumption that I is a model of $\Sigma(\mathcal{A})$. So S_I is *conflict-free*.

2. For admissibility:

$\Rightarrow$ Consider a set S of $\mathcal{A}$ that is *admissible*. An Herbrand interpretation I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ can be defined as in the proof of Proposition 11.1.

With this definition, $S_I = S$. It remains to prove that I is a model of formulas $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$.

As for the proof of Proposition 11.1, I is a model of formulas (4.12) and (4.13).

Let prove that I is a model of $\Sigma_{Def}(\mathcal{A})$. Obviously I is a model of formulas (4.1), (4.2) and (4.3). Since I is a model of formulas (4.13), we must prove that I is a model of formulas (4.4), $(4.5)_{\text{AF}}$, (4.6), (4.7) and $(4.8)_{\text{AF}}$. By definition of I, formulas (4.4), $(4.5\text{b})_{\text{AF}}$ and (4.6) are obviously satisfied. The proof that I also satisfies Formula $(4.5\text{a})_{\text{AF}}$ is the same that the one given for Proposition 11.1.

Consider now Formula (4.7a). Since $\mathcal{A}$ is an AF-C, $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Moreover I is a model of $\Sigma(\mathcal{A})$, and we have seen previously that I satisfies Formula (4.6). And, by definition of I, for any $\dot{e}$ in D_I, $I(Cand(e)) = \top$ iff $I(Arg(e)) = \top$, so, I is a model of the formula $\forall x(Cand(x) \leftrightarrow Arg(x))$. Thus, I is a model of the formula $\forall x(Cand(x) \rightarrow Arg(x))$, and also of the formula $\forall x(Cand(x) \rightarrow Arg(x) \vee Att(x))$. So, following Proposition 7, I is a model of Formula (4.7a).

Consider Formula (4.7b). Let $\dot{e} \in D_I$ such that $I(Cand(e)) = \top$. By definition of I, for any $\dot{e_1}$ in D_I, $I(Cand(e_1)) = \top$ iff $I(Arg(e_1)) = \top$, for any $\dot{e_2}$ in D_I, $I(PrimaFacie(e_2)) = \top$ iff

$e_2 \in \mathscr{P}$ and for any $\dot{e_3}$ in D_I, $I(Unsupportable(e_3) = \bot$. So $I(Unsupportable(e)) = \bot$ and $I(Arg(e)) = \top$. By definition of I, this implies that $e \in \mathscr{A}$. But, since $\mathcal{A}$ is an AF-C, $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. So, $e \in \mathscr{P}$ and $I(PrimaFacie(e)) = \top$. Thus I is not a model of the formula:
$\neg PrimaFacie(e) \wedge \forall y \in Sup(T(y, e_1) \rightarrow Desactivated(y))$.
And so, I is a model of the formula:
$Unsupportable(e) \leftrightarrow$
$[\neg PrimaFacie(e) \wedge \forall y \in Sup(T(y, e_1) \rightarrow Desactivated(y))]$.

Consider Formula $(4.8)_{\text{AF}}$. Assume that I does not satisfy Formula $(4.8)_{\text{AF}}$. So there exists $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$, $I(Acceptable(e_1)) = \top$ and I does not satisfy the formula $\forall y \in Att(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Selected(v)\ )\ )\ )\ )$. By definition of I, and since $I(Acceptable(e_1)) = \top$, $I(Selected(e_1)) = \top$. There also exists $\dot{e_2} \in D_I$ such that $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ with $I(Arg(e_3)) = \top$, $I(S(e_2, e_3)) = \bot$ or for any $\dot{e_4} \in D_I$ with $I(Att(e_4)) = \top$, $I(T(e_4, e_3)) = \bot$ or there exists $\dot{e_5} \in D_I$ such that $I(Arg(e_5)) = \top$, $I(S(e_4, e_5)) = \top$ and $I(Selected(e_5)) = \bot$. By definition of I and since I is a model of Axiom (4.12), $e_1 \in \mathscr{A}$, $e_1 \in S$ and there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, and for any $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$, for any $e_4 \in \mathscr{R}$ with $t(e_4) = e_3$, there exists $e_5 \in \mathscr{A}$ such that $e_5 \in s(e_4)$ and $e_5 \notin S_I$. Thus there exist $e_1 \in S$ and $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and there does not exist $e_4 \in \mathscr{R}$ with $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S$. So, e_1 is not *acceptable* wrt S. That contradicts the assumption that S is *admissible*.

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$. We must prove that S_I is an *admissible* set. Following the proof of Proposition 11.1, we know that S_I is *conflict-free*. It remains to prove that $\forall x \in S_I$, x is *acceptable* wrt S_I. Let $e_1 \in S_I$. Since $S_I \subseteq \mathscr{A}$ and I is a model of formulas (4.1), $e_1 \in \mathscr{A}$ and there exists $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$. By definition of S_I, $I(Selected(e_1)) = \top$, so, since I is a model of Formula (4.4), $I(Acceptable(e_1)) = \top$ and $I(Supported(e_1)) = \top$. Since I is a model of Formula $(4.8)_{\text{AF}}$, I satisfies the formula $\forall y \in Att(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Selected(v)\ )\ )\ )\ )$. Since I is a model of formulas (4.1), (4.2) and (4.3), for any $e_2 \in \mathscr{R}$ such that $e_1 \in t(e_2)$, there exists $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$, and there exists $e_4 \in \mathscr{R}$ such that $e_3 \in t(e_4)$ and for any $e_5 \in \mathscr{A}$ with $e_5 \in s(e_4)$, $e_5 \in S_I$. Since I is a model of Axiom (4.12), e_1 and e_3 are unique. So for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$ and there exists $e_4 \in \mathscr{R}$ such that $t(e_4) = e_3$ and $s(e_4) \subseteq S_I$. This means that for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_4 \in \mathscr{R}$ such that $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S_I$. So, e_1 is *acceptable* wrt S_I and so S_I is *admissible*.

3. For the complete semantics:

$\Rightarrow$ Consider a *complete* extension S of $\mathcal{A}$. An Herbrand interpretation I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ can be defined as follows:

- For any $\dot{e_1}$ in D_I, $I(Acceptable(e_1)) = \top$ iff $\forall e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, $\exists e_3 \in \mathscr{R}$ with $t(e_3) \in s(e_2)$ and $s(e_3) \subseteq S$
- For the other predicates, I is defined as in the proof of Proposition 11.2.

With this definition, $S_I = S$. It remains to prove that I is a model of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$.
As for the proof of Proposition 11.1, I is a model of formulas (4.12) and (4.13).
Let prove that I is a model of $\Sigma_{Rein}(\mathcal{A})$. Obviously, I is a model of formulas (4.1), (4.2) and (4.3). Since I is a model of formulas (4.13), we must prove that I is a model of formulas (4.4), $(4.5)_{\text{AF}}$, (4.6), (4.7), $(4.8)_{\text{AF}}$ and $(4.9)_{\text{AF}}$. By definition of I, Formula $(4.5\text{b})_{\text{AF}}$ is satisfied by I. The proofs that I satisfies formulas $(4.5\text{a})_{\text{AF}}$, (4.6) and (4.7) are identical to the ones given in the proof of Proposition 11.2.

Consider Formula (4.4).

$\Rightarrow$ Let $\dot{e_1} \in D_I$ such that $I(Selected(e_1)) = \top$. By definition of I, $I(Supported(e_1)) = \top$ and $e_1 \in S$. But, S is a *complete* extension, so S is *admissible* and for any x such that x is *acceptable* wrt S, $x \in S$; this implies that e_1 is *acceptable* wrt S. So $\forall e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, $\exists e_3 \in \mathscr{R}$ such that $t(e_3) \in s(e_2)$ and $s(e_3) \subseteq S$. Thus, $I(Acceptable(e_1)) = \top$.

$\Leftarrow$ Let $\dot{e_1} \in D_I$ such that $I(Supported(e_1)) = \top$ and $I(Acceptable(e_1)) = \top$. By definition of I, $\forall e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, $\exists e_3 \in \mathscr{R}$ such that $t(e_3) \in s(e_2)$ and $s(e_3) \subseteq S$. This implies that e_1 is *acceptable* wrt S. But, S is a *complete* extension, so S is *admissible* and for any x such that x is *acceptable* wrt S, $x \in S$. Thus $e_1 \in S$, and so $I(Selected(e_1)) = \top$.

Consider Formula $(4.8)_{\text{AF}}$. Assume that I does not satisfy Formula $(4.8)_{\text{AF}}$. So there exists $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$, $I(Acceptable(e_1)) = \top$ and I does not satisfy the formula $\forall y \in Att(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Selected(v)\ )\ )\ )\ )$. By definition of I, $\forall e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, $\exists e_4 \in \mathscr{R}$ with $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S$. Hence, there also exists $\dot{e_2} \in D_I$ such that $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ with $I(Arg(e_3)) = \top$, $I(S(e_2, e_3)) = \bot$ or for any $\dot{e_4} \in D_I$ with $I(Att(e_4)) = \top$, $I(T(e_4, e_3)) = \bot$ or there exists $\dot{e_5} \in D_I$ such that $I(Arg(e_5)) = \top$, $I(S(e_4, e_5)) = \top$ and $I(Selected(e_5)) = \bot$. By definition of I and because I is a model of Axiom (4.12), there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, and for any $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$, for any $e_4 \in \mathscr{R}$ with $t(e_4) = e_3$, there exists $e_5 \in \mathscr{A}$ such that $e_5 \in s(e_4)$ and $e_5 \notin S_I$. So there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and there does not exist $e_4 \in \mathscr{R}$ with $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S$. This contradicts the fact that $\forall e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, $\exists e_4 \in \mathscr{R}$ with $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S$.

Consider Formula $(4.9)_{\text{AF}}$. Let $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$ and I satisfies the formula $\forall y \in Att(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Selected(v)\ )\ )\ )\ )$. Let show that $I(Acceptable(e_1)) = \top$. By definition of I, $e_1 \in \mathscr{A}$. Moreover, as for the $\Leftarrow$ part of the proof of Proposition 11.2, one can deduce that for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_3 \in \mathscr{R}$ such that $t(e_3) \in s(e_2)$ and $s(e_3) \subseteq S_I$. So $I(Acceptable(e_1)) = \top$.

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$. Following the proof of Proposition 11.2, we know that S_I is *admissible*. So it remains to prove that for any x such that x is *acceptable* wrt S_I, $x \in S_I$.

Consider $e_1 \in \mathscr{A}$ such that e_1 is *acceptable* wrt S_I and let prove that $I(Selected(e_1)) = \top$. Since I is a model of Formula (4.4), it is enough to show that $I(Supported(e_1)) = \top$ and $I(Acceptable(e_1)) = \top$. Since e_1 is *acceptable* wrt s_I, for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_4 \in \mathscr{R}$ such that $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S_I$. Moreover, since I is a model of formulas (4.1), we have $I(Arg(e_1)) = \top$.

Let show that $I(Supported(e_1)) = \top$. By definition of I, I is a model of $\Sigma(\mathcal{A})$. Moreover, since I is a model of formulas (4.13), I is also a model of $\forall x(Cand(x) \leftrightarrow Arg(x))$. This implies that I is a model of the formula $\forall x(Cand(x) \rightarrow Arg(x))$, and also a model of the formula $\forall x(Cand(x) \rightarrow Arg(x) \vee Att(x))$. Following Proposition 6, I is so a model of the formula $\forall x \in Cand(Supported(x))$. But, we know that $I(Arg(e_1)) = \top$, so, since I is a model of $\forall x(Cand(x) \leftrightarrow Arg(x))$, $I(Cand(e_1)) = \top$. Consequently, $I(Supported(e_1)) = \top$.

Let show that $I(Acceptable(e_1)) = \top$. We know that for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_4 \in \mathscr{R}$ such that $t(e_4) \in s(e_2)$ and $s(e_4) \subseteq S_I$. So for any $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$, there exists $e_3 \in \mathscr{A}$ such that $e_3 \in s(e_2)$ and there exists $e_4 \in \mathscr{R}$ such that $t(e_4) = e_3$ and for any $e_5 \in \mathscr{A}$ with $e_5 \in s(e_4)$, $e_5 \in S_I$. Since I is a model of formulas (4.1), (4.2) and (4.3), for any $\dot{e_2} \in D_I$ such that $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$, there exists $\dot{e_3} \in D_I$ such that $I(Arg(e_3)) = \top$

and $I(S(e_2, e_3)) = \top$, and there exists $\dot{e_4} \in D_I$ such that $I(Att(e_4)) = \top$, $I(T(e_4, e_3)) = \top$ and for any $\dot{e_5} \in D_I$ with $I(Arg(e_5)) = \top$ and $I(S(e_4, e_5)) = \top$, $I(Selected(e_5)) = \top$. Thus, I is a model of the formula $\forall y \in Att(T(y, e_1) \rightarrow \exists z \in Arg(S(y, z) \wedge \exists u \in Att(T(u, z) \wedge \forall v \in Arg(S(u, v) \rightarrow Selected(v)))))$. Following Formula $(4.9)_{\text{AF}}$ that is satisfied by I, one can conclude that $I(Acceptable(e_1)) = \top$.

4. For the preferred semantics: Let I be an interpretation of a set of formulas Σ. It is obvious to see that I is a $\subseteq$-*maximal* model of Σ iff S_I is $\subseteq$-*maximal* among the extensions S_J, where J is a model of Σ. Considering $\Sigma = \Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$, one can see that the *preferred* extensions correspond to the extensions S_I where I is a $\subseteq$-*maximal* model of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.12), (4.13)\}$.

5. For the grounded semantics: Let I be an interpretation of a set of formulas Σ. It is obvious to see that I is a $\subseteq$-*minimal* model of Σ iff S_I is $\subseteq$-*minimal* among the extensions S_J, where J is a model of Σ. By definition, the *grounded* extension is the *complete* extension that is $\subseteq$-*minimal*. This implies that the *grounded* extension is the extension S_I where I is a $\subseteq$-*minimal* model of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.12), (4.13)\}$.

6. For the stable semantics:

   $\Rightarrow$ Consider a *stable* extension S of $\mathcal{A}$. An Herbrand interpretation I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.12), (4.13)\}$ can be defined as in the proof of Proposition 11.1.

   With this definition, $S_I = S$. Let prove that I is a model of formulas $\Sigma_{CA}(\mathcal{A}) \cup \{(4.12), (4.13)\}$.

   As for the proof of Proposition 11.1, I is a model of formulas (4.12) and (4.13).

   Let show that I is a model of $\Sigma_{CA}(\mathcal{A})$. Obviously I is a model of formulas (4.1), (4.2) and (4.3). Since I is a model of formulas (4.13), let prove that I is a model of formulas (4.4), $(4.5)_{\text{AF}}$, (4.6), (4.7) and $(4.10)_{\text{AF}}$. By definition of I, formulas (4.4), $(4.5\text{b})_{\text{AF}}$ and (4.6) are obviously satisfied by I. Moreover, the proofs that I satisfies formulas $(4.5\text{a})_{\text{AF}}$ and (4.7) are identical to those given in the proof of Proposition 11.2.

   Consider Formula $(4.10\text{a})_{\text{AF}}$. Let $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$ and $I(Acceptable(e_1)) = \bot$. By definition of I, $I(Selected(e_1)) = \bot$ and so $e_1 \notin S$. But S is a *stable* extension, so S attacks e_1. Thus, there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and $s(e_2) \subseteq S$. So there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and for any $e_3 \in \mathscr{A}$ such that $e_3 \in s(e_2)$, $e_3 \in S$. Then, since I is a model of formulas (4.1), (4.2) and (4.3), there exists $\dot{e_2} \in D_I$ with $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ such that $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Selected(e_3)) = \top$. So, I is a model of the formula $\exists y \in Att(T(y, e_1) \wedge \forall z \in Att(S(y, z) \rightarrow Selected(z)))$. This implies that I is a model of Formula $(4.10\text{a})_{\text{AF}}$.

   Consider Formula (4.10b). Since $\mathcal{A}$ is an AF-C, $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$. Moreover I is a model of $\Sigma(\mathcal{A})$. And by definition of I, for any $\dot{e}$ in D_I, $I(Cand(e)) = \top$ iff $I(Arg(e)) = \top$, so I is a model of the formula $\forall x(Cand(x) \leftrightarrow Arg(x))$. Thus I is a model of the formula $\forall x(Cand(x) \rightarrow Arg(x))$, and also a model of the formula $\forall x(Cand(x) \rightarrow Arg(x) \vee Att(x))$. Following Proposition 9, I is a model of Formula (4.10b).

   $\Leftarrow$ Let I be a Herbrand model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.12), (4.13)\}$. Following the proof of Proposition 11.1, we know that S_I is *conflict-free*. It remains to prove that S_I attacks any $x \in \mathscr{A} \setminus S_I$. Let $e_1 \in \mathscr{A} \setminus S_I$. Since I is a model of formulas (4.1), there exists $\dot{e_1} \in D_I$ such that $I(Arg(e_1)) = \top$. Moreover, by definition of S_I, $I(Selected(e_1)) = \bot$. Since I is a model of Formula (4.4), $I(Acceptable(e_1)) = \bot$. Following Formula $(4.10\text{a})_{\text{AF}}$ that is satisfied by I, I is a model of the formula $\exists y \in Att(T(y, a) \wedge \forall z \in Arg(S(y, z) \rightarrow Selected(z)))$. Thus, there exists $\dot{e_2} \in D_I$ with $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ such that $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Selected(b)) = \top$. Since I is a model of formulas (4.1), (4.2) and (4.3), this implies that there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and for any $e_3 \in \mathscr{A}$ such that $e_3 \in s(e_2)$, $e_3 \in S_I$. So there exists $e_2 \in \mathscr{R}$ such that $t(e_2) = e_1$ and $s(e_2) \subseteq S_I$, and so, S_I attacks e_1.

□

B.6 Proofs for Section 4.5.5: Theory for HO-AF

Proposition. 12 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an HO-AF and $U = (S, \Gamma, \Delta)$ be a structure.*

1. *U is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
2. *U is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
3. *U is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
4. *U is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
5. *U is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
6. *U is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.15)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*

Proof. (for Proposition 12) The idea is to apply a succession of operations on the language used in this work to retrieve the language that is used in the "Logical Description of a RAF" and "Logical Formalization of RAF semantics" sections of [CL20]. In the following, when we mention formulas from [CL20], we refer to formulas given in these particular sections. These operations will then be applied successively to the formulas (4.4) and $(4.5)_{\text{HOAF}}$ to $(4.10)_{\text{HOAF}}$ in order to eventually yield the formulas used in [CL20].

First operation (denoted Θ): correctly rename some of the predicates of the language as follows.

- Arg becomes Arg
- Att becomes $Attack$
- Sup becomes $\bot$ (Sup has no corresponding predicate in [CL20] and in an HO-AF, we have $\mathscr{S} = \varnothing$)
- $PrimaFacie$ becomes $\top$ ($PrimaFacie$ has no corresponding predicate in the logical encoding proposed by [CL20] for a RAF; moreover, in an HO-AF, we consider that $\mathscr{S} = \varnothing$ and $\mathscr{P} = \mathscr{A} \cup \mathscr{R}$)
- $Acceptable$ becomes $Acceptable$ (we will see later that this predicate can be replaced by others)
- $Supported$ becomes $\top$ (Sup has no corresponding predicate in [CL20] and we can use Proposition 6)
- $Unsupportable$ becomes $\bot$ (Sup has no corresponding predicate in [CL20] and we can use Proposition 8)
- $Unacceptable$ becomes $NAcc$
- $Cand$ becomes $\top$ ($Cand$ does not appear in formulas (4.4) and $(4.5)_{\text{HOAF}}$ to $(4.10)_{\text{HOAF}}$ and has no corresponding predicate in [CL20])
- $Activable$ becomes $\top$ (same reason as for $Cand$))
- $Defeated$ becomes $\top$ (same reason as for $Cand$)
- $Inhibited$ becomes $\top$ (same reason as for $Cand$)

- *Desactivated* becomes $\top$ (same reason as for $Cand$)

Once this operation is applied, we separate formulas $\Theta((4.5)_{\text{HOAF}})$ to $\Theta((4.10)_{\text{HOAF}})$ in two groups of subformulas. In each formula, exactly one quantifier occurs that is bounded to $Arg \cup Attack$. The separation consists of putting in the first group the formulas with this quantifier bounded to only Arg (group A) and in the second group the formulas with this quantifier bounded to $Attack$ (group B).

Second operations (denoted Λ): correctly rename the missing unary predicates of the language as follows.

- $Selected(x)$ becomes $Acc(x)$ when $Arg(x)$ is true
- $Selected(x)$ becomes $Val(x)$ when $Attack(x)$ is true

Third operation (denoted Π): replace the use of binary predicates S and T by the introduction of functional terms s_α and t_α (justified by the presence of the axioms (4.11) and (4.12) in the theory).

- $\forall a \in Arg(S(\alpha, a) \rightarrow \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $\exists a \in Arg(S(\alpha, a) \wedge \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $T(\alpha, x)$ becomes $t_\alpha = x$

The idea is then to apply successively Θ, Λ and Π on formulas (4.4) and $(4.5)_{\text{HOAF}}$ to $(4.10)_{\text{HOAF}}$ so that we obtain the formulas used in [CL20]. However, for some formulas, this result is not immediate. We will therefore use different versions of these formulas (namely (4.4), $(4.5\text{a})_{\text{HOAF}}$ and $(4.8)_{\text{HOAF}}$) on which to apply Θ, Λ and Π.

Concerning Formula (4.4), we transform the universal quantifier into a quantifier restricted to Arg and Att. This addition is correct because in Proposition 12 we consider models of theories that contain (4.4), (4.1) and (4.2), and we consider the case of HO-AF in which we have $\mathscr{S} = \varnothing$. Thus, instead of (4.4), we consider Formula $(4.4)_{\text{Diff}}$.

$$\forall x \in (Arg \cup Att)(Selected(x) \leftrightarrow (Acceptable(x) \wedge Supported(x))) \qquad ((4.4)_{\text{Diff}})$$

Concerning Formula $(4.5\text{a})_{\text{HOAF}}$, we change the subformula $\exists x \in (Arg \cup Att)\ \big(T(\alpha, x) \wedge Unacceptable(x)\big)$ into $\forall x \in (Arg \cup Att)\big(T(\alpha, x) \rightarrow Unacceptable(x)\big)$. This modification is valid because in Proposition 12 we consider models of theories that contain (4.12). We then use standard modifications that preserve logical equivalence to put the quantifier $\forall x \in (Arg \cup Att)$ at the beginning of the formula. This results in Formula $(4.5\text{a})_{\text{Diff}}$ which will be used instead of $(4.5\text{a})_{\text{HOAF}}$.

$$\begin{aligned} &\forall x \in (Arg \cup Att)\Big(\forall \alpha \in Att\Big(\\ &\Big[\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge Selected(\alpha) \wedge T(\alpha, x)\Big] \rightarrow \\ &\qquad Unacceptable(x)\ \Big)\Big) \end{aligned} \qquad ((4.5\text{a})_{\text{Diff}})$$

Concerning Formula $(4.8)_{\text{HOAF}}$, we use the same standard modifications that preserve logical equivalence to put the quantifier $\forall \alpha \in Att$ at the beginning of the formula. This results in Formula $(4.8)_{\text{Diff}}$ which will be used instead of $(4.8)_{\text{HOAF}}$.

$$\forall \alpha \in Att\Big(\forall x \in (Arg \cup Att)\Big(\ [Acceptable(x) \wedge T(\alpha, x)] \rightarrow$$
$$\exists \beta \in Att\Big(\ \big[\exists a \in Arg\big(\ S(\alpha, a) \wedge T(\beta, a)\ \big) \vee T(\beta, \alpha)\big] \wedge \qquad ((4.8)_{\text{Diff}})$$
$$\big[\forall b \in Arg\big(\ S(\beta, b) \rightarrow Selected(b)\ \big) \wedge Selected(\beta)\big]\ \Big)\Big)\Big)$$

By applying successively Θ, Λ and Π on formulas $(4.4)_{\text{Diff}}$, $(4.5\text{a})_{\text{Diff}}$, $(4.5\text{b})_{\text{HOAF}}$, $(4.8)_{\text{Diff}}$, $(4.9)_{\text{HOAF}}$, $(4.10)_{\text{HOAF}}$, we obtain the following formulas.

$$\forall x \in Arg(Acc(x) \leftrightarrow (Acceptable(x) \wedge \top)) \qquad ((4.4)_{\text{ShiftA}})$$

$$\forall x \in Attack(Val(x) \leftrightarrow (Acceptable(x) \wedge \top)) \qquad ((4.4)_{\text{ShiftB}})$$

The result of these two previous formulas is that the predicate *Acceptable* becomes equivalent to the predicate *Acc* for arguments and to the predicate Val for attacks. Thus, we obtain:

$$\forall x \in Arg(\forall \alpha \in Attack([Acc(s_\alpha) \wedge Val(\alpha) \wedge (t_\alpha = x)] \rightarrow NAcc(x))) \qquad ((4.5\text{a})_{\text{ShiftA}})$$

$$\forall x \in Attack(\forall \alpha \in Attack([Acc(s_\alpha) \wedge Val(\alpha) \wedge (t_\alpha = x)] \rightarrow NAcc(x))) \qquad ((4.5\text{a})_{\text{ShiftB}})$$

$$\forall x \in Arg(NAcc(x) \rightarrow \neg Acc(x)) \qquad ((4.5\text{b})_{\text{ShiftA}})$$

$$\forall x \in Attack(NAcc(x) \rightarrow \neg Val(x)) \qquad ((4.5\text{b})_{\text{ShiftB}})$$

$$\forall \alpha \in Attack(\forall x \in Arg([Acc(x) \wedge (t_\alpha = x)] \rightarrow \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge Acc(s_\beta) \wedge Val(\beta)))) \qquad ((4.8)_{\text{ShiftA}})$$

$$\forall \alpha \in Attack(\forall x \in Attack([Val(x) \wedge (t_\alpha = x)] \rightarrow \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge Acc(s_\beta) \wedge Val(\beta)))) \qquad ((4.8)_{\text{ShiftB}})$$

$$\forall x \in Arg(\forall \alpha \in Attack((t_\alpha = x) \rightarrow \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge Acc(s_\beta) \wedge Val(\beta))) \rightarrow Acc(x)) \qquad ((4.9)_{\text{ShiftA}})$$

$$\begin{aligned}\forall x \in Attack(\forall \alpha \in Attack((t_\alpha = x) \rightarrow \\ \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge Acc(s_\beta) \wedge Val(\beta))) \\ \rightarrow Val(x))\end{aligned} \qquad ((4.9)_{\text{ShiftB}})$$

$$\forall x \in Arg(\neg Acc(x) \rightarrow \exists \alpha \in Attack(t_\alpha = x \wedge Acc(s_\alpha) \wedge Val(\alpha))) \qquad ((4.10\text{a})_{\text{ShiftA}})$$

$$\forall x \in Attack(\neg Val(x) \rightarrow \exists \alpha \in Attack(t_\alpha = x \wedge Acc(s_\alpha) \wedge Val(\alpha))) \qquad ((4.10\text{a})_{\text{ShiftB}})$$

We have the following immediate results (where "amounts" means "logically equivalent").

- Formula $(4.5\text{a})_{\text{ShiftA}}$ amounts to Formula (2) of [CL20]
- Formulas $(4.5\text{a})_{\text{ShiftB}}$ and $(4.5\text{b})_{\text{ShiftB}}$ together amount to Formula (1) of [CL20]
- Formula $(4.5\text{b})_{\text{ShiftA}}$ amounts to Formula (3) of [CL20]
- Formula $(4.8)_{\text{ShiftA}}$ amounts to Formula (11) of [CL20]
- Formula $(4.8)_{\text{ShiftB}}$ amounts to Formula (12) of [CL20]
- Formula $(4.9)_{\text{ShiftA}}$ amounts to Formula (13) of [CL20]
- Formula $(4.9)_{\text{ShiftB}}$ amounts to Formula (14) of [CL20]
- Formula $(4.10\text{a})_{\text{ShiftA}}$ amounts to Formula (15) of [CL20]
- Formula $(4.10\text{a})_{\text{ShiftB}}$ amounts to Formula (16) of [CL20]

The only missing formulas are those that are used to describe the graph of an argumentation framework, namely (4), (5), (6), (7), (8), (9) and (10) in [CL20]. They should be retrieved using formulas (4.1), (4.2) and (4.3). Here are the details.

Let us consider formulas $(4.1)_{\text{Shift}}$, $(4.2)_{\text{Shift}}$ and $(4.3)_{\text{Shift}}$, which are the formulas obtained by applying Θ on formulas (4.1), (4.2) and (4.3). We have the following results.

- Formula $(4.2\text{a})_{\text{Shift}}$ amounts to Formula (5) of [CL20]
- Formulas $(4.1\text{a})_{\text{Shift}}$ and $(4.2\text{b})_{\text{Shift}}$ together amount to Formula (7) of [CL20]
- Formulas $(4.1\text{b})_{\text{Shift}}$ and $(4.2\text{c})_{\text{Shift}}$ together amount to Formula (8) of [CL20]
- Formula $(4.1\text{d})_{\text{Shift}}$ amounts to formulas (9) and (10) together of [CL20]

Since in Proposition 12 we consider models of theories that contain (4.11) and (4.12), it is obvious that (4.3) can be rewritten as follows.

$$\text{for all } \alpha \in \mathscr{R} \text{ with } s(\alpha) = a \text{ and } t(\alpha) = b, S(\alpha, a) \wedge T(\alpha, b) \qquad ((4.3)_{\text{bis}})$$

Modifying $(4.3)_{\text{bis}}$ by replacing the predicates S and T by the functional terms s_α and t_α allows us to retrieve Formula (6) of [CL20].

To retrieve Formula (4) of [CL20], we use formulas (4.1a) and (4.1b) of our approach. The first issue is that formulas (4.1a) and (4.1b) range over some elements of the argumentation framework, while Formula (4) of [CL20] is universal. However, by formulas (4.2) that are satisfied by the models considered in Proposition 12 and the fact that we consider Herbrand models, the sets over which formulas (4.1a) and (4.1b) range form a partition of the model's domain. Thus, we can gather formulas (4.1a) and (4.1b) into a single formula that uses an unbounded universal quantifier, as follows.

$$\forall x([Arg(x) \wedge \neg Att(x) \wedge \neg Sup(x)] \vee [\neg Arg(x) \wedge Att(x) \wedge \neg Sup(x)]) \qquad ((4.1)_{\text{ab}})$$

One can observe that Formula (4) of [CL20] is the closure of a Formula in CNF (provided we turn the implication into a disjunction) while Formula $(4.1)_{\text{ab}}$ is the closure of a formula in DNF. Let us then compute the CNF of the formula of which $(4.1)_{\text{ab}}$ is the closure. By applying the distributivity property, we obtain the following formula.

$$\begin{aligned}\forall x([Arg(x) \vee Att(x)] \wedge [Arg(x) \vee \neg Sup(x)] \wedge \\ [\neg Att(x) \vee \neg Arg(x)] \wedge [\neg Att(x) \vee \neg Sup(x)] \wedge \\ [\neg Sup(x) \vee \neg Arg(x)] \wedge [\neg Sup(x) \vee Att(x)] \wedge \neg Sup(x))\end{aligned}$$

The previous formula can be simplified by removing the conjuncts that contain two terms amongst which is $\neg Sup(x)$, because the last conjunct consists of $\neg Sup(x)$. This gives the following formula.

$$\forall x([Arg(x) \vee Att(x)] \wedge [\neg Att(x) \vee \neg Arg(x)] \wedge \neg Sup(x))$$

If we apply Θ to the previous formula, we obtain formulas (4) and (5) of [CL20] ($\neg Sup(x)$ becomes $\top$ and can thus be removed).

In conclusion, since our theory is equivalent to the theory given in [CL20], Proposition 4.1 of [CL20] and Proposition 12 are also equivalent. And so Proposition 12 holds. □

B.7 Proofs for Section 4.5.6: Theory for EBAF

In this section, some additional definitions, propositions and lemmas are useful in order to prove the main proposition concerning the translation of EBAFs.

B.7.1 Additional Definitions

These definitions are another, more set-theoretic, way to define the usual semantics for EBAF as given in Section 2.3.3. We prefer to use these definitions as we find them easier to handle. We first prove that they are indeed equivalent to the original definitions, and then proceed to use them in order to prove the main result of this section, namely Proposition 13.

Note that some of these definitions are those from Section 4.4, which have been modified to correspond to EBAF using the language operations (†) and (‡). There are also new definitions that are not from Section 4.4 but that are convenient when using a set-theoretic approach.

Definition 78 (Defeat)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An element $x \in \mathscr{A}$ is *defeated* by S if and only if there exists $\alpha \in \mathscr{R}$ such that $s(\alpha) \in S$ and $t(\alpha) = x$.

Notation. We recall that the set of all elements defeated by S is denoted $Def(S)$.

Remark. $Def(S) \stackrel{\text{def}}{=} \{a \in \mathscr{A} \mid \exists \alpha \in \mathscr{R}, s(\alpha) \in S \text{ and } t(\alpha) = a\}$.

Definition 79 (Support)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An element $x \in \mathscr{A}$ is *e-supported* by S if and only if $x \in \mathscr{P}$ or there exists $\alpha \in \mathscr{S}$ with (1) $t(\alpha) = x$, (2) $s(\alpha) \in S$ and (3) $s(\alpha)$ is *e-supported* by $S' = S \setminus \{x\}$.

Notation. We recall that the set of all elements e-supported by S is denoted $Supp(S)$.

Remark. $Supp(S) \stackrel{\text{def}}{=} \mathscr{P} \cup \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\}$.

Definition 80 (Unsupportability)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An element $x \in \mathscr{A}$ is *unsupportable* by S if and only if x is *not e-supported* by the set of arguments $S' = \mathscr{A} \setminus Def(S)$.

Notation. We recall that the set of all elements unsupportable by S is denoted $UnSupp(S)$.

Remark. $UnSupp(S) \stackrel{\text{def}}{=} (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$.

Definition 81 (Unacceptability)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An element $x \in \mathscr{A}$ is *unacceptable* with respect to S if and only if (1) x is defeated by S or (2) x is unsupportable by S.

Notation. In the following, considering a set of arguments S, we denote by $UnAcc(S)$ the set of all elements unacceptable with respect to S.

Remark. $UnAcc(S) \stackrel{\text{def}}{=} Def(S) \cup UnSupp(S)$.

Definition 82 (Unactivability)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An attack $\alpha \in \mathscr{R}$ is *unactivable* with respect to a set S if and only if $s(\alpha)$ is *unacceptable* with respect to S.

Notation. In the following, considering a set of arguments S, we denote by $UnAct(S)$ the set of all elements unactivable with respect to S.

Remark. $UnAct(S) \stackrel{\text{def}}{=} \{\alpha \in \mathscr{R} \mid s(\alpha) \in UnAcc(S)\}$.

Definition 83 (Acceptability)**.** Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF and S a set of arguments. An element $x \in \mathscr{A}$ is *acceptable* with respect to a S if and only if (1) x is e-supoprted by S and (2) for all $\alpha \in \mathscr{R}$, if $t(\alpha) = x$, then $s(\alpha)$ is either defeated by S or unsupportable by S.

Notation. We recall that the set of all elements acceptable with respect to S is denoted $Acc(S)$.

Remark. $Acc(S) \stackrel{\text{def}}{=} \{a \in \mathscr{A} \mid a \in Supp(S)$ and $\forall \alpha \in \mathscr{R}$ st $t(\alpha) = a, s(\alpha) \in UnAct(S)\}$.

Definition 84 (Usual semantics)**.** A set $S \subseteq \mathscr{A}$ is:

- *self-supported* iff $S \subseteq Supp(S)$,
- *conflict-free* iff $S \cap Def(S) = \varnothing$,
- *admissible* iff S is *conflict-free* and $S \subseteq Acc(S)$,
- *complete* iff S is *conflict-free* and $S = Acc(S)$,
- *preferred* iff S is a $\subseteq$-*maximal admissible* set,
- *grounded* iff S is a $\subseteq$-*minimal complete* set,
- *stable* iff $S = \mathscr{A} \setminus UnAcc(S)$.

B.7.2 Additional Propositions and Lemmas for Correspondence of Definitions

We now proceed to prove that the semantics captured using the additional definitions are the same as the usual semantics.

Lemma 14. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. An argument a is e-supported by S iff $a \in Supp(S)$.*

Proof. (for Lemma 14) Obvious following Definitions 14 and 79. □

Lemma 15. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. For any T s.t. $S \subseteq T$, $Supp(S) \subseteq Supp(T)$.*

Remark. A variant of this lemma is Lemma A.1 given for REBAF in [CFFL18a]

Proof. (for Lemma 15) The proof is made by induction on the number of elements in S.

Initial step: If $S = \varnothing$, then, by Definition 79, $Supp(S) = \mathscr{P} \cup \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\} = \mathscr{P}$. So for any T s.t. $S \subseteq T$, $\mathscr{P} \subseteq \mathscr{P} \cup \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in T \cap Supp(T \setminus \{t(\alpha)\})\}$.

Induction step: Let $S_n \subseteq \mathscr{A}$ be a set whose size is n. Let $a \in Supp(S_n)$ and T_n s.t. $S_n \subseteq T_n$. By Definition 79, $a \in \mathscr{P} \cup \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in S_n \cap Supp(S_n \setminus \{t(\alpha)\})\}$. If $a \in \mathscr{P}$, the reasoning is similar to the initial step. So let assume that $a \in \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in S_n \cap Supp(S_n \setminus \{t(\alpha)\})\}$. Since $S_n \subseteq T_n$, $(S_n \cap Supp(S_n \setminus \{t(\alpha)\})) \subseteq (T_n \cap Supp(S_n \setminus \{t(\alpha)\}))$.

- If $a \in S_n$, then $S_{n-1} = S_n \setminus \{a\}$ is a set whose size is $n-1$. The induction hypothesis applies and for any T s.t. $S_{n-1} \subseteq T$, $Supp(S_{n-1}) \subseteq Supp(T)$. As $S_n \subseteq T_n$, $S_{n-1} \subseteq T_n \setminus \{a\}$. So, $Supp(S_n \setminus \{a\}) \subseteq Supp(T_n \setminus \{a\})$, and thus, $(T_n \cap Supp(S_n \setminus \{a\})) \subseteq (T_n \cap Supp(T_n \setminus \{a\}))$. So $a \in \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in T_n \cap Supp(T_n \setminus \{t(\alpha)\})\}$, and this implies that $a \in Supp(T_n)$.
- If $a \notin S_n$, $(S_n \setminus \{a\}) = S_n$. But, $a \in \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in S_n \cap Supp(S_n \setminus \{t(\alpha)\})\}$. So, $s(\alpha) \in S_n$ and $s(\alpha) \in Supp(S_n)$. Since $a \notin S_n$ and $S_n \subseteq T_n$, we can see that $S_n \subseteq (T_n \setminus \{a\})$. So $s(\alpha) \in T_n$ and $s(\alpha) \in T_n \setminus \{a\}$. With a reasoning similar to the previous one (replacing a by $s(\alpha)$ and T_n by $T_n \setminus \{a\}$), we have $s(\alpha) \in Supp(T_n \setminus \{a\})$. So, $s(\alpha) \in (T_n \cap Supp(T_n \setminus \{a\}))$. Finally we obtain $a \in \{t(\alpha) \mid \exists \alpha \in \mathscr{S}, s(\alpha) \in T_n \cap Supp(T_n \setminus \{t(\alpha)\})\}$, and this implies that $a \in Supp(T_n)$.

□

Lemma 16. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. Let $a \in \mathscr{A}$. $x \in Supp(S)$ iff $x \in Supp(S \setminus \{x\})$.*

Proof. (for Lemma 16)

⇒ Let $x \in Supp(S)$. By Definition 79, $x \in \mathscr{P} \cup \{t(\alpha)\ \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\}$. Two cases are possible.

- If $x \in \mathscr{P}$, then, by definition, $x \in Supp(S \setminus \{x\})$.
- If $x \in \{t(\alpha)\ \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\}$, two subcases are possible.
  - If $x = s(\alpha)$, by definition, $s(\alpha) \in S \cap Supp(S \setminus \{x\})$, so $x \in Supp(S \setminus \{x\})$.
  - If $x \neq s(\alpha)$, one can note that $s(\alpha) \in S \setminus \{x\}$. Moreover $(S \setminus \{x\}) \setminus \{x\} = S \setminus \{x\}$. So $s(\alpha) \in (S \setminus \{x\}) \cap Supp((S \setminus \{x\}) \setminus \{x\})$. Thus, by Definition 79, $x \in Supp(S \setminus \{x\})$.

⇐ Let $x \in Supp(S \setminus \{x\})$. Since $(S \setminus \{x\}) \subseteq S$, by Lemma 15, we have $Supp(S \setminus \{x\}) \subseteq Supp(S)$. Since $x \in Supp(S \setminus \{x\})$, then $x \in Supp(S)$.

□

Proposition 15. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. S is* self-supported$_1$ *(in the sense of Definition 15) iff S is* self-supported$_2$ *(in the sense of Definition 84).*

Proof. (for Proposition 15)

$$\begin{aligned}
S \text{ is } \textit{self-supported}_1 \text{ iff } & \forall a \in S, a \text{ is e-supported by } S && \text{(Definition 15)}\\
\text{iff } & \forall a \in S, a \in Supp(S) && \text{(Lemma 14)}\\
\text{iff } & S \subseteq Supp(S)\\
\text{iff } & S \text{ is } \textit{self-supported}_2 && \text{(Definition 84)}
\end{aligned}$$

□

Lemma 17. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be a* self-supported *extension.*[1]*. S e-attacks an argument a iff $a \in Def(S)$.*

Proof. (for Lemma 17)

⇒ Obvious by Definitions 17 and 78.

⇐ Let assume that $a \in Def(S)$. By Definition 78, $\exists \alpha \in \mathscr{R}$ s.t. $t(\alpha) = a$ and $s(\alpha) \in S$. But, S is a *self-supported* extension. So, by Definition 84, $\forall b \in S$, $b \in Supp(S)$. By Lemma 14, $\forall b \in S$ is e-supported by S. In particular, $s(\alpha) \in S$, so $s(\alpha)$ is e-supported by S. So, by Definition 17, S e-attacks a.

□

Proposition 16. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. S is* conflict-free$_1$ *(in the sense of Def. 19) iff S is* conflict-free$_2$ *(in the sense of Def. 84).*

Proof. (for Proposition 16)

$$\begin{aligned}
S \text{ is } \textit{conflict-free}_1 \text{ iff } & \nexists a, b \in S \text{ s.t. } \exists \alpha \in \mathscr{R} \text{ with } s(\alpha) = a \text{ and } t(\alpha) = b && \text{(Def. 19)}\\
\text{iff } & \nexists b \in S \text{ s.t. } \exists \alpha \in \mathscr{R} \text{ with } s(\alpha) \in S \text{ and } t(\alpha) = b\\
\text{iff } & \nexists b \in S \text{ s.t. } b \in Def(S) && \text{(Def. 78)}\\
\text{iff } & S \cap Def(S) = \varnothing\\
\text{iff } & S \text{ is } \textit{conflict-free}_2 && \text{(Def. 84)}
\end{aligned}$$

□

Lemma 18. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be a* conflict-free *extension.*[2]*. Then, $UnSupp(S) \subseteq (\mathscr{A} \setminus Supp(S))$.*

Proof. (for Lemma 18) Since S is *conflict-free*, by Definition 84, we have $S \cap Def(S) = \varnothing$. Since $S \cap Def(S) = \varnothing$ and $S \subseteq \mathscr{A}$, $S \subseteq (\mathscr{A} \setminus Def(S))$. So, by Lemma 15, $Supp(S) \subseteq Supp(\mathscr{A} \setminus Def(S))$. So, $\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)) \subseteq (\mathscr{A} \setminus Supp(S))$, and as $UnSupp(S) \stackrel{\text{def}}{=} (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$ by Definition 80, $UnSupp(S) \subseteq (\mathscr{A} \setminus Supp(S))$. □

Lemma 19. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. An argument a is unsupportable wrt S iff for any set $T \subseteq \mathscr{A}$ s.t. $a \in Supp(T)$, $T \cap Def(S) \neq \varnothing$.*

Proof. (for Lemma 19)

⇒ By Definition 80, $a \in (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$. So, $a \notin Supp(\mathscr{A} \setminus Def(S))$. Let assume that $T \cap Def(S) = \varnothing$. Since $T \subseteq \mathscr{A}$, we can deduce that $T \subseteq (\mathscr{A} \setminus Def(S))$. Moreover, $a \in Supp(T)$, so, by Lemma 15, $a \in Supp(\mathscr{A} \setminus Def(S))$. We obtain a contradiction, and so $T \cap Def(S) \neq \varnothing$.

[1] Following Proposition 15, it is not useful to make a distinction between *self-supported*$_1$ and *self-supported*$_2$.
[2] By Proposition 16, it is not useful to make a distinction between *conflict-free*$_1$ and *conflict-free*$_2$.

$\Leftarrow$ Let assume that $a \notin UnSupp(S)$. By Definition 80, this means that $a \notin (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$. Since $a \in \mathscr{A}$, we can deduce that $a \in Supp(\mathscr{A} \setminus Def(S))$. But, $(\mathscr{A} \setminus Def(S)) \subseteq \mathscr{A}$ and $(\mathscr{A} \setminus Def(S)) \cap Def(S) = \varnothing$. This contradicts the assumption that, for any set $T \subseteq \mathscr{A}$ s.t. $a \in Supp(T)$, $T \cap Def(S) \neq \varnothing$. So, $a \in UnSupp(S)$

□

Lemma 20. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be a* self-supported *extension. An argument a is e-acceptable wrt S iff $a \in Acc(S)$.*

Proof. (for Lemma 20)

$\Rightarrow$ Let a be an argument e-acceptable wrt S. By Definition 18, a is e-supported by S, so following Lemma 14, $a \in Supp(S)$. Consider $\alpha \in \mathscr{R}$ s.t. $t(\alpha) = a$. We must prove that $\alpha \in UnAct(S)$.

- If $\nexists T \subseteq \mathscr{A}$ s.t. $s(\alpha) \in Supp(T)$, then $\forall T \subseteq \mathscr{A}$, $s(\alpha) \in (\mathscr{A} \setminus Supp(T))$. In particular, $s(\alpha) \in (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$. So, $s(\alpha) \in UnSupp(S)$. By Definition 81, $s(\alpha) \in UnAcc(S)$, and so, by Definition 82, $\alpha \in UnAct(S)$.
- Assume now that $\exists T \subseteq \mathscr{A}$ s.t. $s(\alpha) \in Supp(T)$. By Lemma 14, this means that $s(\alpha)$ is e-supported by T, and so, by Definition 17, $T \cup \{s(\alpha)\}$ e-attacks a. By Definition 18, this implies that S e-attacks an element in $T \cup \{s(\alpha)\}$, and so, by Definition 17, $\exists \beta \in \mathscr{R}$ s.t. $t(\beta) \in (T \cup \{s(\alpha)\})$, $s(\beta) \in S$ and $s(\beta)$ is e-supported by S.
  - Assume that $t(\beta) = s(\alpha)$. By Definition 78, this means that $s(\alpha) \in Def(S)$.
  - Moreover, assume now that $t(\beta) \neq s(\alpha)$. So, $t(\beta) \in T$, and thus $T \cap Def(S) \neq \varnothing$.

  So, either $s(\alpha) \in Def(S)$, or $\forall T \subseteq \mathscr{A}$ s.t. $s(\alpha) \in Supp(T)$, $T \cap Def(S) \neq \varnothing$. So, by Lemma 19, either $s(\alpha) \in Def(S)$, or $s(\alpha) \in UnSupp(S)$. Thus, by Definitions 81 and 82, $\alpha \in UnAct(S)$.

$\Leftarrow$ Let $a \in Acc(S)$. By Definition 83, we have $a \in Supp(S)$ and $\forall \alpha \in \mathscr{R}$ s.t. $t(\alpha) = a$, $\alpha \in UnAct(S)$. By Lemma 14, a is e-supported by S. By Definition 82, $s(\alpha) \in UnAcc(S)$, and so, by Definition 81, $s(\alpha) \in Def(S) \cup UnSupp(S)$.

- Assume that $s(\alpha) \in Def(S)$. By Lemma 17, S e-attacks $s(\alpha)$. Consider $T \subseteq \mathscr{A}$ s.t. T e-supports $s(\alpha)$. By Definition 17, $T \cup \{s(\alpha)\}$ e-attacks a, and, since S e-attacks $s(\alpha)$, S e-attacks an element in $T \cup \{s(\alpha)\}$. Since a is e-supported by S, by Definition 18, a is e-acceptable wrt S.
- Assume now that $s(\alpha) \in UnSupp(S)$. Consider $T \subseteq \mathscr{A}$ s.t. T e-supports $s(\alpha)$. By Definition 17, $X = T \cup \{s(\alpha)\}$ e-attacks a. By Lemma 14, $s(\alpha) \in Supp(T)$. By Lemma 19, this means that $T \cap Def(S) \neq \varnothing$. By Definition 78 and Lemma 17, S e-supports an attack against an element in T. So, S e-supports an attack against an element in X. So, for any set X s.t. X e-supports an attacks against a, S e-supports an attacks against an element in X. So, by Definition 18, a is e-acceptable wrt S.

□

Lemma 21. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be an* admissible$_1$ *extension (in the sense of Def. 4). So, S is* self-supported.

Proof. (for Lemma 21) By Definition 4, $\forall a \in S$ a is e-acceptable wrt S. So, by Definition 18, $\forall a \in S$, a is e-supported by S. Thus, by Definition 15, S is *self-supported*. □

Lemma 22. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be an* admissible$_2$ *extension (in the sense of Def. 84). Then, S is* self-supported.

Remark. A variant of this lemma is Lemme A.6 given for REBAF in [CFFL18a]

Proof. (for Lemma 22) By Definition 84, $S \subseteq Acc(S)$. So, by Definition 83, $\forall a \in S$, $a \in Supp(S)$. So, $S \subseteq Supp(S)$. Thus, by Definition 84, S is *self-supported*. □

Proposition 17. *Let* $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ *be an EBAF. Let* $S \subseteq \mathscr{A}$. *S is* admissible$_1$ *(in the sense of Def. 4) iff S is* admissible$_2$ *(in the sense of Def. 84).*

Proof. (for Proposition 17)

S is *admissible*$_1$	iff S is *conflict-free*$_1$ and $\forall a \in S, a$ is e-acceptable wrt S	(Def. 4)
	iff S is *conflict-free*$_2$ and $\forall a \in S, a$ is e-acceptable wrt S	(Prop. 16)
	iff S is *conflict-free*$_2$ and $\forall a \in S, a \in Acc(S)$	(Lem. 21 and 20)
	iff S is *conflict-free*$_2$ and $S \subseteq Acc(S)$	
	iff S is *admissible*$_2$	(Def. 84)

□

Proposition 18. *Let* $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ *be an EBAF. Let* $S \subseteq \mathscr{A}$. *S is* complete$_1$ *(in the sense of Def. 4) iff S is* complete$_2$ *(in the sense of Def. 84).*

Proof. (for Proposition 18)

S is *complete*$_1$	iff S is *admissible*$_1$ and	
	$\forall a \in \mathscr{A}$ s.t. a is e-acceptable wrt $S, a \in S$	(Def. 4)
	iff S is *admissible*$_1$ and $\forall a \in \mathscr{A}$ s.t. $a \in Acc(S), a \in S$	(Lem. 21 and 20)
	iff S is *admissible*$_1$ and $Acc(S) \subseteq S$	
	iff S is *admissible*$_2$ and $Acc(S) \subseteq S$	(Prop. 17)
	iff S is *conflict-free*$_2, S \subseteq Acc(S)$ and $Acc(S) \subseteq S$	(Def. 84)
	iff S is *conflict-free*$_2$ and $Acc(S) = S$	
	iff S is *complete*$_2$	(Def. 84)

□

Proposition 19. *Let* $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ *be an EBAF. Let* $S \subseteq \mathscr{A}$. *S is* preferred$_1$ *(in the sense of Def. 4) iff S is* preferred$_2$ *(in the sense of Def. 84).*

Proof. (for Proposition 19)

S is *preferred*$_1$	iff S is an *admissible*$_1$ $\subseteq$-*maximal* set	(Def. 4)
	iff S is an *admissible*$_2$ $\subseteq$-*maximal* set	(Prop. 17)
	iff S is *preferred*$_2$	(Def. 84)

□

Proposition 20. *Let* $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ *be an EBAF. Let* $S \subseteq \mathscr{A}$. *S is* grounded$_1$ *(in the sense of Prop. 1) iff S is* grounded$_2$ *(in the sense of Def. 84).*

Proof. (for Proposition 20)

S is *grounded*$_1$	iff S is a *complete*$_1$ $\subseteq$-*minimal* set	(Prop. 1)
	iff S is a *complete*$_2$ $\subseteq$-*minimal* set	(Prop. 18)
	iff S is *grounded*$_2$	(Def. 84)

□

Lemma 23. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. $(\mathscr{A} \setminus S) \subseteq UnAcc(S)$ iff $\forall a \in (\mathscr{A} \setminus S)$ s.t. a is e-supported by $\mathscr{A}$, $\exists \alpha \in \mathscr{R}$ with (1) $s(\alpha) \in S$ and (2) $t(\alpha) = a$ or $t(\alpha) \in T$ for any set T s.t. a is minimally e-supported by T.*

Proof. (for Lemma 23)

⇒ Assume that $(\mathscr{A} \setminus S) \subseteq UnAcc(S)$. Let $a \in (\mathscr{A} \setminus S)$ s.t. a is e-supported by $\mathscr{A}$. Since $a \in UnAcc(S)$, by Definition 81, $a \in Def(S) \cup UnSupp(S)$.

- Assume that $a \in Def(S)$. By Definition 78, $\exists \alpha \in \mathscr{R}$ s.t. $s(\alpha) \in S$ and $t(\alpha) = a$.
- Assume now that $a \in UnSupp(S)$. By Lemma 19, for any set $T \subseteq \mathscr{A}$ s.t. $a \in Supp(T)$, $T \cap Def(S) \neq \varnothing$. By Lemma 14 and Definition 78, this means that for any set $T \subseteq \mathscr{A}$ s.t. a is e-supported by T, $\exists \alpha \in \mathscr{R}$ s.t. $s(\alpha) \in S$ and $t(\alpha) \in T$. Since it holds for any set $T \subseteq \mathscr{A}$ s.t. a is e-supported by T, then it also holds for any set $T \subseteq \mathscr{A}$ s.t. a is *minimally* e-supported by T.

⇐ Assume that $\forall a \in (\mathscr{A} \setminus S)$ s.t. a is e-supported by $\mathscr{A}$, $\exists \alpha \in \mathscr{R}$ with (1) $s(\alpha) \in S$ and (2) $t(\alpha) = a$ or $t(\alpha) \in T$ for any set T s.t. a is minimally e-supported by T. Let $a \in (\mathscr{A} \setminus S)$ s.t. a is e-supported by $\mathscr{A}$. We must prove that $a \in UnAcc(S)$. By assumption, there exists $\alpha \in \mathscr{R}$ with (1) $s(\alpha) \in S$ and (2) $t(\alpha) = a$ or $t(\alpha) \in T$ for any set T s.t. a is minimally e-supported by T.

- If $t(\alpha) = a$ then, by Definition 78, $a \in Def(S)$.
- Assume now that $t(\alpha) \in T$ for any set T s.t. a is minimally e-supported by T. By Definition 78, this implies that for any set T s.t. a is minimally e-supported by T, $T \cap Def(S) \neq \varnothing$. Let $ESupMin(a)$ be the set of sets $T \subseteq \mathscr{A}$ s.t. a is minimally e-supported by T. Let T' s.t. a is e-supported by T'. As such, $\nexists T \in ESupMin(a)$ s.t. $T' \subset T$. So, $\exists T \in ESupMin(a)$ s.t. $T \subseteq T'$. However, $\forall T \in ESupMin(a)$, $T \cap Def(S) \neq \varnothing$, so $T' \cap Def(S) \neq \varnothing$. Thus, for any set T' s.t. a is e-supported by T', $T' \cap Def(S) \neq \varnothing$. By Lemma 14, for any set T' s.t. $a \in Supp(T')$, $T' \cap Def(S) \neq \varnothing$. By Lemma 19, this implies that $a \in UnSupp(S)$.

Finally, since $a \in Def(S)$ or $a \in UnSupp(S)$, then $a \in Def(S) \cup UnSupp(S)$, and so, By Definition 81, $a \in UnAcc(S)$.

□

Lemma 24. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ be a* self-supported *and* conflict-free *set. Then, $UnAcc(S) \subseteq (\mathscr{A} \setminus S)$.*

Proof. (for Lemma 24) Assume that S is *self-supported* and *conflict-free*. Consider $a \in UnAcc(S)$. By Definition 81, $a \in (Def(S) \cup UnSupp(S))$. Two cases are possible:

- If $a \in Def(S)$, since S is *conflict-free*, we have by Definition 84 $S \cap Def(S) = \varnothing$ and since $S \subseteq \mathscr{A}$, $a \in (\mathscr{A} \setminus S)$.
- If $a \in UnSupp(S)$, by Definition 80, $a \in (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$. We must prove that $(\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S))) \subseteq (\mathscr{A} \setminus S)$. Since S is *conflict-free*, $S \cap Def(S) = \varnothing$, and since $S \subseteq \mathscr{A}$, $S \subseteq (\mathscr{A} \setminus Def(S))$. By Lemma 15, we have $Supp(S) \subseteq Supp(\mathscr{A} \setminus Def(S))$. Moreover, since S is *self-supported*, by Definition 84, $S \subseteq Supp(S)$. So we have $S \subseteq Supp(\mathscr{A} \setminus Def(S))$. Considering the complement sets wrt $\mathscr{A}$ of these sets, we obtain that $(\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S))) \subseteq (\mathscr{A} \setminus S)$.

□

Lemma 25. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ s.t. $S = \mathscr{A} \setminus UnAcc(S)$. Let S_2 s.t. $S_2 \subseteq (\mathscr{A} \setminus Def(S))$. Let $S_1 = S \cap S_2$. Then, $S_2 \cap Supp(S_2) \subseteq S_1 \cap Supp(S_1)$.*

Proof. (for Lemma 25) Consider $a \in S_2 \cap Supp(S_2)$. The proof is split into 2 steps, the first one for proving that $a \in S_1$ and the second one for proving that $a \in Supp(S_1)$.

- Since $a \in S_2 \cap Supp(S_2)$, $a \in Supp(S_2)$. But, $S_2 \subseteq (\mathscr{A} \setminus Def(S))$, so $a \in \mathscr{A} \setminus Def(S)$ and, by Lemma 15, $Supp(S_2) \subseteq Supp(\mathscr{A} \setminus Def(S))$, and so, $a \in Supp(\mathscr{A} \setminus Def(S))$. So, $a \in (\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S))$. But, $(\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S)) = \mathscr{A} \setminus (Def(S) \cup (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S))))$. Moreover, by Definition 81, $\mathscr{A} \setminus (Def(S) \cup (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))) = \mathscr{A} \setminus UnAcc(S)$. And finally, by assumption, $\mathscr{A} \setminus UnAcc(S) = S$. So, $a \in S$, and as $a \in S_2$, we have $a \in S \cap S_2$ and consequently $a \in S_1$.

- The proof for $a \in Supp(S_1)$ is done by induction on S_2.

  Initial step: If $S_2 = \varnothing$, then, obviously, $S_2 \cap Supp(S_2) \subseteq Supp(S_1)$.

  Induction step: Assume that for any $S_2' \subset S_2$, $S_2' \cap Supp(S_2') \subseteq Supp(S_1')$, with $S_1' = S \cap S_2'$. By Lemma 16, since $a \in Supp(S_2)$, $a \in Supp(S_2 \setminus \{a\})$. As $a \in S_2$, we have $(S_2 \setminus \{a\}) \subset S_2$. Let denote $S_2 \setminus \{a\}$ by S_2' and apply the induction hypothesis. Thus, $S_1' = S_1 \setminus \{a\}$. Since $a \in Supp(S_2')$, by Definition 79, $a \in \mathscr{P} \cup \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S_2' \cap Supp(S_2')\}$. If $a \in \mathscr{P}$, then by definition, $a \in Supp(S_1)$, so we assume that $\exists \alpha \in \mathscr{S}$ s.t. $t(\alpha) = a$ and $s(\alpha) \in S_2' \cap Supp(S_2')$. By the induction hypothesis, we have so $s(\alpha) \in Supp(S_1')$, and by a proof similar to that used in the first step, $s(\alpha) \in S_1'$. Since $s(\alpha) \in S_1'$, $s(\alpha) \in S_1$. So, $\exists \alpha \in \mathscr{S}$ s.t. $t(\alpha) = a$ and $s(\alpha) \in S_1 \cap Supp(S_1')$. Thus, by Definition 79, $a \in Supp(S_1)$.

□

Lemma 26. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$ s.t. $UnAcc(S) = \mathscr{A} \setminus S$. Then, S is* conflict-free *and* self-supported.

Proof. (for Lemma 26) Proof in two parts, the first one for showing that S is *conflict-free* and the second one for proving that S is *self-supported*.

- Assume that $UnAcc(S) = \mathscr{A} \setminus S$. So, $S = \mathscr{A} \setminus UnAcc(S)$. By Definition 81, $UnAcc(S) = Def(S) \cup UnSupp(S)$. So, $\mathscr{A} \setminus UnAcc(S) = \mathscr{A} \setminus (Def(S) \cup (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))) = (\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S))$. So, $S \subseteq \mathscr{A} \setminus Def(S)$ and since $Def(S) \subseteq \mathscr{A}$, we can deduce that $S \cap Def(S) = \varnothing$. Thus, by Definition 84, S is *conflict-free*.

- Moreover, since following Definition 81, we have $S = (\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S))$, it is obvious that $(\mathscr{A} \setminus Def(S)) \subseteq (\mathscr{A} \setminus Def(S))$. We apply Lemma 25 with $S_2 = (\mathscr{A} \setminus Def(S))$. And we obtain that $S_2 \cap Supp(S_2) \subseteq S_1 \cap Supp(S_1)$ with $S_1 = S \cap S_2$. But, since $S = (\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S))$, $S \subseteq (\mathscr{A} \setminus Def(S))$, so, $S_1 = S \cap (\mathscr{A} \setminus Def(S)) = S$. Moreover, since $S = (\mathscr{A} \setminus Def(S)) \cap Supp(\mathscr{A} \setminus Def(S))$ and $S_2 = (\mathscr{A} \setminus Def(S))$, we have $S = S_2 \cap Supp(S_2$. Thus, $S \subseteq S \cap Supp(S)$, and so, $S \subseteq Supp(S)$. Then, by Definition 84, S is *self-supported*.

□

Proposition 21. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. S is* stable_1 *(in the sense of Def. 20) iff S is* stable_2 *(in the sense of Def. 84).*

Proof. (for Proposition 21)

S is $stable_1$ iff S is *conflict-free*$_1$, *self-supported*$_1$, and
$\forall a \in (\mathscr{A} \setminus S)$ s.t. a is e-supported by $\mathscr{A}$,
$\exists \alpha \in \mathscr{R}$ with (1) $s(\alpha) \in S$ and (2) $t(\alpha) = a$ or
$t(\alpha) \in T$ for any set T s.t. a is minimally
e-supported by T. (Def. 20)
iff S is *conflict-free*$_1$, *self-supported*$_1$, and
$(\mathscr{A} \setminus S) \subseteq UnAcc(S)$ (Lem. 23)
iff S is *conflict-free*$_2$, *self-supported*$_2$, and
$(\mathscr{A} \setminus S) \subseteq UnAcc(S)$ (Prop. 16 and 15)
iff $UnAcc(S) \subseteq (\mathscr{A} \setminus S)$ and $(\mathscr{A} \setminus S) \subseteq UnAcc(S)$ (Lem. 24 and 26)
iff $UnAcc(S) = (\mathscr{A} \setminus S)$
iff $S = (\mathscr{A} \setminus UnAcc(S))$
iff S is $stable_2$ (Def. 84)

□

Lemma 27. *Let $(\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF. Let $S \subseteq \mathscr{A}$. Let $x \in \mathscr{A} \setminus \mathscr{P}$. Then, $x \in UnSupp(S)$ iff for any $\alpha \in \mathscr{S}$ s.t. $t(\alpha) = x$, $s(\alpha) \in UnSupp(S)$ or $s(\alpha) \in Def(S)$.*

Proof. (for Lemma 27)

- ⇒ By Lemma 19, we know that for any set $T \subseteq \mathscr{A}$ s.t. $x \in Supp(T)$, $T \cap Def(S) \neq \varnothing$. But, $x \notin \mathscr{P}$, so, $x \notin Supp(\varnothing)$ and so, by Definition 14, for any set $T \subseteq \mathscr{A}$ s.t. $x \in Supp(T)$, $T = T' \cup \{s(\alpha)\}$ with $T' \subseteq \mathscr{A}$, $\alpha \in \mathscr{S}$, $t(\alpha) = x$ and $s(\alpha) \in Supp(T')$. Thus, for any set $T' \cup \{s(\alpha)\}$ with $T' \subseteq \mathscr{A}$, $\alpha \in \mathscr{S}$, $t(\alpha) = x$ and $s(\alpha) \in Supp(T')$, $(T' \cup \{s(\alpha)\}) \cap Def(S) \neq \varnothing$. Consequently, for any $\alpha \in \mathscr{S}$ s.t. $t(\alpha) = x$, $s(\alpha) \in Def(S)$ or for any T' with $T' \subseteq \mathscr{A}$ and $s(\alpha) \in Supp(T')$, $T' \cap Def(S) \neq \varnothing$. So, by Lemma 19, for any $\alpha \in \mathscr{S}$ s.t. $t(\alpha) = x$, $s(\alpha) \in Def(S)$ or $s(\alpha) \in UnSupp(S)$.
- ⇐ By Lemma 19, we have, for any $\alpha \in \mathscr{S}$ s.t. $t(\alpha) = x$, $s(\alpha) \in Def(S)$ or $\forall T' \subseteq \mathscr{A}$ s.t. $s(\alpha) \in Supp(T')$, $T' \cap Def(S) \neq \varnothing$. Thus, for any $\alpha \in \mathscr{S}$ s.t. $t(\alpha) = x$, for any $T' \subseteq \mathscr{A}$ s.t. $s(\alpha) \in Supp(T')$, $(T' \cup \{s(\alpha)\}) \cap Def(S) \neq \varnothing$. But, we know that $x \notin \mathscr{P}$. Consequently, $x \notin Supp(\varnothing)$. Thus, for any $T \subseteq \mathscr{A}$ s.t. $x \in Supp(T)$, we know that T is under the form of $T' \cup \{s(\alpha)\}$ with $T' \subseteq \mathscr{A}$ and $\alpha \in \mathscr{S}$ and $t(\alpha) = x$. So, for any $T \subseteq \mathscr{A}$ s.t. $x \in Supp(T)$, $T \cap Def(S) \neq \varnothing$. By Lemma 19, this implies that $x \in UnSupp(S)$.

□

B.7.3 Additional Lemmas for the Logical Encoding

The lemmas here will be used as intermediate results so as to make the proof of Proposition 13 a bit more concise.

Lemma 28. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF without support cycle. Let I be a Herbrand model of $\Sigma_{SS}(\mathcal{A}) \cup (4.16) \cup (4.11) \cup (4.12)$. Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(Supported(e_1)) = \top$. Then, $e_1 \in Supp(S_I)$.*

Proof. (for Lemma 28) The proof is made by induction on the size of a specific path in $\mathscr{S}$, this path contains only elements of S_I and is originated in a *prima facie* argument.

Induction Hypothesis: Let C be an $\mathscr{S}$-path, containing only elements of S_I and originated in a *prima facie* argument, the size of this path being n. Then the final element of the path belongs to $Supp(S_I)$.

Initial step: Assume that the size of the path C is 0. So, $C = (e_1)$. e_1 is then the origin and the final element of C. But, we know that the origin of C is a *prima facie* argument, so $e_1 \in \mathscr{P}$, and so, by Definition 79, $e_1 \in Supp(S_I)$.

Induction step: Assume that the size of C is n. If $I(PrimaFacie(e_1)) = \top$, by formulae (4.1) satisfied by I, we obtain that $e_1 \in \mathscr{P}$, and we conclude that $e_1 \in Supp(S_I)$ as in the initial step. Assume that $I(PrimaFacie(e_1)) = \bot$. Since $I(Supported(e_1)) = \top$, by Formula $(4.7a)_{\text{EBAF}}$ satisfied by I, this implies that I is a model of the formula $\exists y \in Supp(T(y, e_1) \wedge \forall z \in Arg(S(y, z) \rightarrow Acc(z)))$. Since I is is a model of formulae (4.1), (4.2) and (4.3), this means that there exists $\dot{e_2} \in D_I$ s.t. $e_2 \in \mathscr{S}$, $t(e_2) = e_1$ and for any $\dot{e_3} \in D_I$ s.t. $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Acc(e_3)) = \top$. Since I is also a model of Axiom (4.11), e_3 is in fact unique, and so, $e_3 = s(e_2)$. Moreover, by Definition of S_I, we know that $e_3 \in S_I$. Thus, there exists $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$ and $s(e_2) \in S_I$. Furthermore, $I(Acc(e_3)) = \top$ and I is a model of Formula (4.4), so, $I(Supported(e_3)) = \top$. Consider $C = (p_f, \ldots, e_3, e_1)$. The size of C is n, so the size of the path $C' = (p_f, \ldots, e_3)$ is $n - 1$. In addition, since C' is is sub-path of C, C' is a path containing only elements of S_I and whose final element is e_3. So the induction hypothesis can be applied and $e_3 \in Supp(S_I)$. Finally, since there is no support cycles in $\mathcal{A}$, e_1 cannot belong to C', so we have $e_3 \in Supp(S_I \setminus \{e_1\})$. Thus, $e_3 \in S_I \cap Supp(S_I \setminus \{e_1\})$. Since $e_3 = s(e_2)$, $s(e_2) \in S_I \cap Supp(S_I \setminus \{e_1\})$, and so, by Definition 79, $e_1 \in Supp(S_I)$. □

Lemma 29. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF without support cycle. Let I be a Herbrand model of $\Sigma_{SS}(\mathcal{A}) \cup (4.16) \cup (4.11) \cup (4.12)$. Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(Unsupportable(e_1)) = \top$. Then $e_1 \in UnSupp(S_I)$.*

Proof. (for Lemma 29) Let consider the inverse relation of the support, *i.e.* the functions s' and t' s.t. $\forall \alpha \in \mathscr{S}, s'(\alpha) = t(\alpha)$ and $t'(\alpha) = s(\alpha)$, and $\forall \alpha \in \mathscr{R}, s'(\alpha) = s(\alpha)$ and $t'(\alpha) = t(\alpha)$. Any argument of $\mathcal{A}$ is then the root of a tree in this new relation (the height of this tree might be equal to 0). The proof of this lemma is made by induction on the height of this tree.

Induction Hypothesis: Let a tree in the inverse relation of the support, whose height is n and whose root is such that there exists an element $\dot{e_1} \in D_I$ representing it with $I(Unsupportable(e_1)) = \top$. Then $e_1 \in UnSupp(S_I)$.

Initial step: Assume that the height of the tree is 0. The tree is so reduced to its root e_1. So there exists no $e_2 \in \mathscr{S}$ s.t. $s'(e_2) = e_1$. So, by definition of s', there exists no $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$. Since I satisfies Formula $(4.7b)_{\text{EBAF}}$ and $I(Unsupportable(e_1)) = \top$, we have $I(PrimaFacie(e_1)) = \bot$. Since I is a model of formulae (4.1), we can deduce that $e_1 \notin \mathscr{P}$. Since there exists no $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$ and $e_1 \notin \mathscr{P}$, then there also exists no $T \subseteq \mathscr{A}$ s.t. $e_1 \in Supp(T)$. Thus, for any set $T \subseteq \mathscr{A}$, $e_1 \in \mathscr{A} \setminus Supp(T)$. In particular, for $T = \mathscr{A} \setminus Def(S_I)$ we obtain that $e_1 \in \mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S_I))$ and so, by Definition 80, $e_1 \in UnSupp(S_I)$.

Induction step: Assume that the height of the tree in the inverse relation of the support is n. Its root is denoted by e_1. As for the initial step, one can deduce that $e_1 \notin \mathscr{P}$. Since I satisfies Formula $(4.7b)_{\text{EBAF}}$, I also satisfies Formula $\forall y \in Supp(\ T(y, e_1) \rightarrow [\exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )]\ )$. Since I is a model of formulae (4.1), (4.2) and (4.3), we obtain that, for any $e_2 \in \mathscr{S}$ s.t. $e_1 \in t(e_2)$, there exists $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$ and either $I(Unsupportable(e_3)) = \top$, or there exists $e_4 \in \mathscr{R}$ s.t. $e_3 \in t(e_4)$ and for any $e_5 \in \mathscr{A}$ s.t. $e_5 \in s(e_4)$, $I(Acc(e_5)) = \top$. Moreover, I is a model of axioms (4.11) and (4.12), so e_1 is the unique target of e_2 and e_3 and e_5 are also unique. Moreover, by definition of S_I, we have $e_5 \in S_I$. So, for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, there exists $e_3 \in \mathscr{A}$ with $s(e_2) = e_3$, and either $I(Unsupportable(e_3)) = \top$, or there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and $s(e_4) \in S_I$.

- If there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and $s(e_4) \in S_I$, by Definition 78, $e_3 \in Def(S_I)$, and since $s(e_2) = e_3$, $s(e_2) \in Def(S_I)$.

- Assume that $I(Unsupportable(e_3)) = \top$. Since $s(e_2) = e_3$ and $t(e_2) = e_1$, by definition of s' and t', we have $s'(e_2) = e_1$ and $t'(e_2) = e_3$. But the height of the tree, whose root is e_1, is n, so, as there is no support cycle in $\mathcal{A}$, the height of the tree whose root is e_3 is strictly less than n. So the induction hypothesis can be applied and since $I(Unsupportable(e_3)) = \top$, $e_3 \in UnSupp(S_I)$ and so, because $s(e_2) = e_3$, $s(e_2) \in UnSupp(S_I)$.

In conclusion, for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, either $s(e_2) \in UnSupp(S_I)$, or $s(e_2) \in Def(S_I)$. So, knowing that $e_1 \notin \mathscr{P}$, one can deduce, by Lemma 27, that $e_1 \in UnSupp(S_I)$. □

Lemma 30. *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an EBAF without support cycle. Let I be a Herbrand model of $\Sigma_{SS}(\mathcal{A}) \cup (4.16) \cup (4.11) \cup (4.12)$. Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $e_1 \in UnSupp(S_I)$. Then, $I(Unsupportable(e_1)) = \top$.*

Proof. (for Lemma 30) Consider the inverse relation of the support, *i.e.* the functions s' and t' s.t. $\forall \alpha \in \mathscr{S}, s'(\alpha) = t(\alpha)$ and $t'(\alpha) = s(\alpha)$, and $\forall \alpha \in \mathscr{R}, s'(\alpha) = s(\alpha)$ and $t'(\alpha) = t(\alpha)$. Any argument of $\mathcal{A}$ is so the root of a tree in this new relation (possibly with an height equals to 0). The proof of the lemma is made by induction on the height of a tree in the inverse relation of the support.

Induction hypothesis: Let a tree in the inverse relation of the support whose height is n and whose root belongs to $UnSupp(S_I)$. Then there exists in D_I an element $\dot{e_1}$ s.t. e_1 is equals to the root of the tree and $I(Unsupportable(e_1)) = \top$.

Initial step: Assume that the height of the tree is 0. The tree is so reduced to its root e_1. So there exists no $e_2 \in \mathscr{S}$ s.t. $s'(e_2) = e_1$. Thus, by definition of s', there exists no $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$. Since $e_1 \in UnSupp(S_I)$, $e_1 \notin \mathscr{P}$. Since I is a model of Formula (4.1), one can deduce that there exists $\dot{e_1} \in D_I$ with $I(Arg(e_1)) = \top$ and $I(PrimaFacie(e_1)) = \bot$. Moreover since I is a model of formulae (4.1), (4.2) and (4.3), one can also deduce that there exists no $\dot{e_2} \in D_I$ s.t. $I(Supp(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$. So, for any $\dot{e_2} \in D_I$ s.t. $I(Supp(e_2)) = \top$, $I(T(e_2, e_1)) = \bot$. Thus, for any formula φ, I is a model of the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \varphi\ )$. In particular, I is a model of the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$. Finally, as $I(PrimaFacie(e_1)) = \bot$ and as I is a model for Formula $(4.7b)_{\text{EBAF}}$, we have that $I(Unsupportable(e_1)) = \top$.

Induction step: Assume that the height of the tree in the inverse relation of the support is n. Its root is denoted by e_1. As for the initial step, one can deduce that there exists $\dot{e_1} \in D_I$ with $I(Arg(e_1)) = \top$ and $I(PrimaFacie(e_1)) = \bot$. By Lemma 27, we know that for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, $s(e_2) \in Def(S_I)$ or $s(e_2) \in UnSupp(S_I)$. Two cases are possibles.

- If $s(e_2) \in Def(S_I)$, there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $e_3 \in Def(S_I)$. By Definition 78, there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and $s(e_4) \in S_I$. So, there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and there exists $e_5 \in \mathscr{A}$ s.t. $s(e_4) = e_5$ and $e_5 \in S_I$. Since I is a model of formulae (4.1), (4.2) and (4.3), this means that for any $\dot{e_2} \in D_I$ s.t. $I(Supp(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$, there exist $\dot{e_3}, \dot{e_4}, \dot{e_5} \in D_I$ with $I(Arg(e_3)) = \top$, $I(Att(e_4)) = \top$, $I(Arg(e_5)) = \top$, $I(S(e_2, e_3)) = \top$, $I(T(e_4, e_3)) = \top$ and $I(S(e_4, e_5)) = \top$. Moreover, by definition of S_I, we have $I(Acc(e_5)) = \top$. Since I is a model of Axiom (4.11), e_5 is unique and so, I satisfies the formula $\exists u \in Att(\ T(u, e_3) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )$. Thus, I satisfies the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$.
- If $s(e_2) \in UnSupp(S_I)$, there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $e_3 \in UnSupp(S_I)$. Since I is a model of formulae (4.1), (4.2) and (4.3), there exist $\dot{e_2}, \dot{e_3} \in D_I$ with $I(Supp(e_2)) = \top$, $I(Arg(e_3)) = \top$, $I(T(e_2, e_1)) = \top$ and $I(S(e_2, e_3)) = \top$. But, since the height of the tree, whose root is e_1, is n, and there is no support cycle in $\mathcal{A}$, then the height of the tree, whose root is e_3, is strictly less that n. So the induction hypothesis can be applied and since $e_3 \in UnSupp(S_I)$, then, $I(Unsupportable(e_3)) = \top$. Thus, I is a model of the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ )\ )$.

In conclusion, either I satisfies the formula $\forall y \in Supp(\ T(y,e_1) \rightarrow \exists z \in Arg\ (\ S(y,z) \wedge \exists u \in Att(\ T(u,z) \wedge \forall v \in Arg(\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )$ or I satisfies the formula $\forall y \in Supp(\ T(y,e_1) \rightarrow \exists z \in Arg(\ S(y,z) \wedge Unsupportable(z)\ )\ )$. So, I satisfies the formula $\forall y \in Supp\ (\ T(y,e_1) \rightarrow (\exists z \in Arg\ (\ S(y,z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg\ (\ S(y,z) \wedge \exists u \in Att\ (\ T(u,z) \wedge \forall v \in Arg\ (\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )\ )$. Moreover, since $I(PrimaFacie(e_1)) = \bot$, $I(Arg(e_1)) = \top$ and I is a model of Formula $(4.7b)_{\text{EBAF}}$, we can conclude that $I(Unsupportable(e_1)) = \top$. □

B.7.4 Proof of the Main Proposition Concerning the Translation of EBAFs

Proposition. 13 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an EBAF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *S is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
2. *S is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
3. *S is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
4. *S is* preferred *if and only if there exists a* $\subseteq$-maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
5. *S is* grounded *if and only if there exists a* $\subseteq$-minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*
6. *S is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ such that $S = S_I \cup \Gamma_I \cup \Delta_I$.*

Proof. (for Proposition 13)

1. For conflict-freeness:

   ⇒ Consider an extension S of $\mathcal{A}$ that is *conflict-free*. We define a Herbrand interpretation I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Supp(e)) = \top$ iff $e \in \mathscr{S}$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1,e_2)) = \top$ iff $e_1 \in \mathscr{R} \cup \mathscr{S}$ and $e_2 \in s(x)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1,e_2)) = \top$ iff $e_1 \in \mathscr{R} \cup \mathscr{S}$ and $e_2 \in t(x)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \top$ iff $e \in \mathscr{P}$
   - For any $\dot{e}$ in D_I, $I(Supported(e)) = \top$ and $I(Unsupportable(e)) = \bot$[3]
   - For any $\dot{e}$ in D_I, $I(Acc(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Cand(e)) = \top$ iff $I(Acc(e)) = \top$
   - For any $\dot{e}$ in D_I, $I(NCand(e)) = \top$ iff $I(Cand(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Element(e)) = \top$ iff $I(Arg(e)) = \top$
   - For any $\dot{e_1}$ in D_I, $I(Activable(e_1)) = \top$ iff for any $\dot{e_2}$ in D_I s.t. $I(Arg(e_2)) = \top$, if $I(S(e_1,e_2)) = \top$ then $I(Acc(e_2)) = \top$
   - For any $\dot{e_1}$ in D_I, $I(Defeated(e_1)) = \top$ iff there exists $\dot{e_2}$ in D_I s.t. $I(Att(e_2)) = \top$, $I(T(e_2,e_1)) = \top$ and $I(Activable(e_2)) = \top$

[3]The support relation is not used in the notion of *conflict-freeness*. In particular, a *conflict-free* extension in an EBAF ($\mathscr{A}$, $\mathscr{R}$, $\mathscr{S}$, $\mathscr{P}$, s, t) is also a *conflict-free* extension in $(\mathscr{A}, \mathscr{R}, \mathscr{S}, s, t, \mathscr{A} \cup \mathscr{R} \cup \mathscr{S})$. So we can consider a model in which any argument is supported.

- For any $\dot{e_1}$ in D_I, $I(Inhibited(e_1)) = \top$ iff there exists $\dot{e_2}$ in D_I s.t. $I(Arg(e_2)) = \top$, $I(S(e_1, e_2)) = \top$ and $I(Defeated(e_2)) = \top$
- For any $\dot{e_1}$ in D_I, $I(Deactivated(e_1)) = \top$ iff $I(Inhibited(e_1)) = \top$ or there exists $\dot{e_2}$ in D_I s.t. $I(Arg(e_2)) = \top$, $I(S(e_1, e_2)) = \top$ and $I(Unsupportable(e_2)) = \top$

With this definition, $S_I = S$. It remains to prove that I is a model of formulae $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$.

By definition of I and since $s : \mathscr{R} \cup \mathscr{S} \mapsto \mathscr{A}$ and $t : \mathscr{R} \cup \mathscr{S} \mapsto \mathscr{A}$ ($\mathcal{A}$ being an EBAF), I is a model of axioms (4.11) and (4.12). Moreover, by definition of I, we also have that I is a model of formulae (4.16).

We must prove that I is a model of $\Sigma_{Coh}(\mathcal{A})$. It is obvious to see that I is a model of formulae (4.1), (4.2) and (4.3). It remains to show that I is is a model of formulae (4.4), (4.5) and (4.6). Since I is a model of formulae (4.16), then we must prove that I is a model of formulae (4.4), $(4.5)_{\text{EBAF}}$ and $(4.6)_{\text{EBAF}}$. By definition of I, formulae (4.4), $(4.5b)_{\text{EBAF}}$ and $(4.6)_{\text{EBAF}}$ are obviously satisfied.

I satisfies Formula $(4.5a)_{\text{EBAF}}$: Assume that I does not satisfy Formula $(4.5a)_{\text{EBAF}}$. So there exists $\dot{e_2} \in D_I$ s.t. $I(Att(e_2)) = \top$, for any $\dot{e_3} \in D_I$ s.t. $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Acc(e_3)) = \top$, and for any $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(T(e_2, e_1)) = \top$, $I(NCand(e_1)) = \top$. By definition of I, for any $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(T(e_2, e_1)) = \top$, $I(Cand(e_1)) = \bot$ and so $I(Acc(e_1)) = \top$. Since I is a model of axioms (4.11) and (4.12), e_1 and e_3 are unique. Thus, by definition of I, $\exists e_2 \in \mathscr{R}$ s.t. $s(e_2) \in S$ and $t(e_2) = e_1$ with $e_1 \in S$. So $e_1 \in Def(S)$ and thus, $S \cap Def(S) \neq \varnothing$, that contradicts the fact that S is *conflict-free* (Definition 84).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$. Assume that S_I is not *conflict-free*. So there exist $e_2 \in \mathscr{R}$ and $e_1 \in \mathscr{A}$ s.t. $s(e_2) \in S_I$, $e_1 \in S_I$, and $t(e_2) = e_1$. Because of formulae (4.1) that are satisfied by I, there exist $\dot{e_2}, \dot{e_1} \in D_I$ s.t. $I(Att(e_2)) = \top$ and $I(Arg(e_1)) = \top$. Similarly, since I is a model of Formula (4.3), we have $I(T(e_2, e_1)) = \top$ and there exists $\dot{e_3} \in D_I$ s.t. $I(S(e_2, e_3)) = \top$. By the definition of S_I, $I(Acc(e_1)) = \top$ and $I(Acc(e_3)) = \top$. Since I is a model of Formula (4.4), we can deduce that $I(Cand(e_1)) = \top$, and so, by Formula $(4.5b)_{\text{EBAF}}$ satisfied by I, $I(NCand(e_1)) = \bot$. Since I is a model of Axiom (4.11), I is a model of the formula $\forall x \in Arg(\ S(e_2, x) \rightarrow Acc(x)\ )$. Similarly, since I is a model of Axiom (4.12), I is a model of the formula $\exists y \in Arg(\ T(e_2, y) \wedge \neg NCand(y)\ )$, that contradicts Formula $(4.5a)_{\text{EBAF}}$, and so the assumption saying that I is a model of $\Sigma(\mathcal{A})$. □

2. For admissibility:

$\Rightarrow$ Consider an extension S of $\mathcal{A}$ that is *admissible*. A Herbrand interpretation I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ can be defined as follows:

- For any $\dot{e}$ in D_I, $I(Supported(e)) = \top$ iff $e \in Supp(S)$
- For any $\dot{e}$ in D_I, $I(Unsupportable(e)) = \top$ iff $e \in UnSupp(S)$
- The interpretation of the other predicates is similar to the one given in the proof of Proposition 13.1

With this definition, $S_I = S$. It remains to prove that I is a model of formulae $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$.

As in the proof of Proposition 13.1, I is a model of formulae (4.11), (4.12) and (4.16).

We must prove that I is a model of $\Sigma_{Def}(\mathcal{A})$. It is obvious that I is a model of formulae (4.1), (4.2) and (4.3). It remains to show that I is a model of formulae (4.4), (4.5), (4.6), (4.7) and (4.8). Since I is a model of formulae (4.16), it is enough to prove that I is a model of formulae (4.4), $(4.5)_{\text{EBAF}}$, $(4.6)_{\text{EBAF}}$, $(4.7)_{\text{EBAF}}$ and $(4.8)_{\text{EBAF}}$. By definition of I, Formula $(4.5b)_{\text{EBAF}}$ is obviously

satisfied. The proof that I satisfies Formula $(4.5a)_{\text{EBAF}}$ is similar to the one given in the proof of Proposition 1.

I satisfies Formula (4.4)**:**

$\Rightarrow$ Let $\dot{e_1} \in D_I$ s.t. $I(Acc(e_1)) = \top$. By the definition of I, $I(Cand(e_1)) = \top$ and $e_1 \in S$. But, S is an *admissible* extension, so, by Definition 84, $e_1 \in Acc(S)$. By Definition 83, $x \in Supp(S)$, and so, by definition of I, we have $I(Supported(e_1)) = \top$.

$\Leftarrow$ Let $\dot{e_1} \in D_I$ s.t. $I(Supported(e_1)) = \top$ and $I(Cand(e_1)) = \top$. By definition of I, we have $I(Acc(e_1)) = \top$.

I satisfies Formula $(4.6)_{\text{EBAF}}$**:** Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$, and I satisfies the formula $PrimaFacie(e_1) \vee \exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(e_2, z) \rightarrow Acc(z)\ )\ )$. We must prove that $I(Supported(e_1)) = \top$. We have $I(PrimaFacie(e_1)) = \top$ or I satisfies the formula $\exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$.

- Assume that $I(PrimaFacie(e_1)) = \top$. So, by definition of I, $e_1 \in \mathscr{P}$, so $e_1 \in Supp(S)$ and $I(Supported(e_1)) = \top$.
- Assume that I satisfies the formula $\exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$. So there exists $\dot{e_2} \in D_I$ such that $I(Supp(e_2)) = \top$, $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ such that $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Acc(e_3)) = \top$. So, by definition of I, $\exists e_2 \in \mathscr{S}$ s.t. $e_1 \in t(e_2)$ and $\forall e_3 \in \mathscr{A}$ s.t. $e_3 \in s(e_2)$, $e_3 \in S$. Since I is a model of axioms (4.11) and (4.12), e_1 and e_3 are unique. As S is *admissible*, by Definition 84, $e_3 \in Acc(S)$, and so, by Definition 83, $e_3 \in Supp(S)$. Two cases are possible.
  - If $e_1 \in S$, by Definition 84, $e_1 \in Acc(S)$, so, by Definition 83, $e_1 \in Supp(S)$, and by definition of I, $I(Supported(e_1)) = \top$.
  - If $e_1 \notin S$, then $(S \setminus \{e_1\}) = S$. So $e_3 \in Supp(S \setminus \{e_1\})$. Thus, $e_3 \in S \cap Supp(S \setminus \{e_1\})$, and since $s(e_2) = e_3$ and $t(e_2) = e_1$, by Definition 79, $e_1 \in Supp(S)$. So, by definition of I, $I(Supported(e_1)) = \top$.

I satisfies Formula $(4.7a)_{\text{EBAF}}$**:** Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(Supported(e_1)) = \top$. We must show that I satisfies the formula $PrimaFacie(e_1) \vee \exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$. By definition of I, $e_1 \in Supp(S)$. So, by Definition 79, $e_1 \in \mathscr{P} \cup \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\}$. Two cases are possible.

- If $e_1 \in \mathscr{P}$, then by definition of I, $I(PrimaFacie(e_1)) = \top$, and so I satisfies the formula $PrimaFacie(e_1) \vee \exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$.
- If $e_1 \in \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S \cap Supp(S \setminus \{t(\alpha)\})\}$, then $\exists e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, $s(e_2) = e_3$ and $e_3 \in S \cap Supp(S \setminus \{e_1\})\}$. So, by definition of I, there exist $\dot{e_2}, \dot{e_3} \in D_I$ s.t. $I(Supp(e_2)) = \top$, $I(Arg(e_3)) = \top$, $I(T(e_2, e_1)) = \top$, $I(S(e_2, e_3)) = \top$ and $I(Acc(e_3)) = \top$. Since I is a model of Axiom (4.11), I satisfies the formula $\forall z \in Arg(\ S(e_2, z) \rightarrow Acc(z)\ )$. Thus, I satisfies the formula $\exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$, and also the formula $PrimaFacie(e_1) \vee \exists y \in Supp(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$.

I satisfies Formula $(4.7b)_{\text{EBAF}}$**:** Consider $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$.

$\Rightarrow$ Assume that $I(Unsupportable(e_1)) = \top$.
We must show that $I(PrimaFacie(e_1)) = \bot$ and that I is a model of the formula $\forall y \in Supp\ (\ T(y, e_1) \rightarrow (\ \exists z \in Arg\ (S(y, z) \wedge Unsupportable(z)) \vee \exists z \in Arg\ (\ S(y, z) \wedge \exists u \in Att\ (T(u, z) \wedge \forall v \in Arg\ (S(u, v) \rightarrow Acc(v)))\ )\ )\ )$. By definition of I, we know that $I(Unsupportable(e_1)) = \top$ implies that $e_1 \in UnSupp(S)$. Then:
  - By Definition 80, $e_1 \in (\mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S)))$. So, $e_1 \notin Supp(\mathscr{A} \setminus Def(S))$. By Definition 79, we can deduce that $e_1 \notin \mathscr{P}$ and so, by definition of I, $I(PrimaFacie(e_1)) = \bot$.

* Since $e_1 \notin \mathscr{P}$, by Lemma 27, we have, for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, $s(e_2) \in Def(S)$ or $s(e_2) \in UnSupp(S)$. By definition of I, we have $I(Supp(e_2)) = \top$ and $I(T(e_2, e_1)) = \bot$. Two cases are possible.
  - If $s(e_2) \in Def(S)$, then, by Definition 78, $\exists e_4 \in \mathscr{R}$ s.t. $t(e_4) = s(e_2)$ and $s(e_4) \in S$. By the definition of I, there exist $\dot{e_3}, \dot{e_4}, \dot{e_5} \in D_I$ s.t. $I(Arg(e_3)) = \top$, $I(Att(e_4)) = \top$, $I(Arg(e_5)) = \top$, $I(S(e_2, e_3)) = \top$, $I(T(e_4, e_3)) = \top$, $I(S(e_4, e_5)) = \top$ and $I(Acc(e_5)) = \top$. Since I is a model of Axiom (4.11), I is a model of the formula $\forall v \in Arg(S(e_4, v) \rightarrow Acc(v))$, and so it is also a model of the formula $\exists z \in Arg\ (\ S(e_2, z) \wedge \exists u \in Att(T(u, z) \wedge \forall v \in Arg(S(u, v) \rightarrow Acc(v)))\ )$.
  - If $s(e_2) \in UnSupp(S)$, then, by the definition of I, there exists $\dot{e_3} \in D_I$ s.t. $I(Arg(e_3)) = \top$, $I(S(e_2, e_3)) = \top$ and $I(Unsupportable(e_3)) = \top$. So I is a model of the formula $\exists z \in Arg(S(e_2, z) \wedge Unsupportable(z))$.

  In conclusion, for any $\dot{e_2} \in D_I$ such that $I(Supp(e_2)) = \top$ and $I(T(e_2, e_1)) = \bot$, I is a model of the formula $\exists z \in Arg(S(e_2, z) \wedge Unsupportable(z)) \vee \exists z \in Arg(\ S(e_2, z) \wedge \exists u \in Att(T(u, z) \wedge \forall v \in Arg(S(u, v) \rightarrow Acc(v)))\ )$. So, I is a model of the formula $\forall y \in Supp\ (\ T(y, e_1) \rightarrow (\ \exists z \in Arg\ (S(y, z) \wedge Unsupportable(z)) \vee \exists z \in Arg\ (\ S(y, z) \wedge \exists u \in Att\ (T(u, z) \wedge \forall v \in Arg\ (S(u, v) \rightarrow Acc(v)))\ )\ )\ )$.

$\Leftarrow$ Assume that $I(PrimaFacie(e_1)) = \bot$ and I is a model of the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow (\ \exists z \in Arg(S(y, z) \wedge Unsupportable(z)) \vee \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(T(u, z) \wedge \forall v \in Arg(S(u, v) \rightarrow Acc(v)))\ )\ )\ )$. Following the first assumption, by definition of I, we can deduce that $e_1 \notin \mathscr{P}$. Following the second assumption, by definition of I and because I is a model of Axiom (4.12), we can deduce that for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, I is a model of the formula $\exists z \in Arg(S(e_2, z) \wedge Unsupportable(z)$ or I is a model of the formula $\exists z \in Arg(\ S(e_2, z) \wedge \exists u \in Att(T(u, z) \wedge \forall v \in Arg(S(u, v) \rightarrow Acc(v)))\ )$.
  * If I is a model of $\exists z \in Arg\ (S(e_2, z) \wedge Unsupportable(z))$, since I is a model of Axiom (4.11), by definition of I, there exists $e_3 \in \mathscr{A}$ s.t. $s(e_2) = e_3$ and $e_3 \in UnSupp(S)$. So, $s(e_2) \in UnSupp(S)$.
  * If I is a model of the formula $\exists z \in Arg\ (\ S(e_2, z) \wedge \exists u \in Att\ (T(u, z) \wedge \forall v \in Arg\ (S(u, v) \rightarrow Acc(v)))\ )$, then by definition of I and because I is a model of axioms (4.11) and (4.12), there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$, there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and there exists $e_5 \in \mathscr{A}$ s.t. $s(e_4) = e_5$ and $e_5 \in S$. So, there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = s(e_2)$ and $s(e_4) \in S$. By Definition 78, we have $s(e_2) \in Def(S)$.

  In conclusion, for any $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$, $s(e_2) \in UnSupp(S)$ or $s(e_2) \in Def(S)$. Since we know that $e_1 \notin \mathscr{P}$, by Lemma 27, we have $e_1 \in UnSupp(S)$ and thus, by definition of I, $I(Unsupportable(e_1)) = \top$.

I **satisfies Formula** $(4.8)_{\text{EBAF}}$**:** Assume that I does not satisfy Formula $(4.8)_{\text{EBAF}}$. So there exists $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$, $I(Cand(e_1)) = \top$ and I does not satisfy the formula $\forall y \in Att(\ T(y, e_1) \rightarrow (\exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$. By definition of I and because I is a model of axioms (4.11) and (4.12), we have $e_1 \in \mathscr{A}$ and there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, and for any $e_3 \in \mathscr{A}$ with $s(e_2) = e_3$, $e_3 \notin UnSupp(S)$ and for any $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$, there exists $e_5 \in \mathscr{A}$ with $s(e_4) = e_5$ and $e_5 \notin S$. So, $e_1 \in \mathscr{A}$ and there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $s(e_2) \notin UnSupp(S)$ and for any $e_4 \in \mathscr{R}$ s.t. $t(e_4) = s(e_2)$, $s(e_4) \notin S$. Donc, by Definition 78, $s(e_2) \notin Def(S)$. Since $s(e_2) \notin UnSupp(S)$, by Definition 81, $s(e_2) \notin UnAcc(S)$ and so, by definition 82, $e_2 \notin UnAct(S)$. But, $I(Cand(e_1)) = \top$, so, by definition of I, $I(Acc(e_1)) = \top$ and thus, $e_1 \in S$. Moreover, S is *admissible*, so by Definition 84, $e_1 \in Acc(S)$; this implies that, by Definition 83, $\forall \alpha \in \mathscr{R}$ s.t. $t(\alpha) = x$, $\alpha \in UnAct(S)$. This contradicts the initial assumption.

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$. We must prove that S_I is an

admissible extension. Following the proof of Proposition 13.1, we know that S_I is a *conflict-free* extension. It remains to prove that $S_I \subseteq Acc(S_I)$. Let $e_1 \in S_I$. As $S_I \subseteq \mathscr{A}$, $e_1 \in \mathscr{A}$, and since I is a model of formulae (4.1), there exists $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$. By definition of S_I, $I(Acc(e_1)) = \top$, so, since I is a model of Formula (4.4), we have that $I(Cand(e_1)) = \top$ and $I(Supported(e_1)) = \top$. Since there is no support cycle in $\mathcal{A}$, and I is a model of formulae $\Sigma_{ss}(AFF) \cup (4.16) \cup (4.11) \cup (4.12)$, and $I(Arg(e_1)) = \top$ and $I(Supported(e_1)) = \top$, by Lemma 28, $e_1 \in Supp(S_I)$. Note that S_I is *self-supported*.

We also know that $I(Cand(e_1)) = \top$. Since I is a model of Formula $(4.8)_{\text{EBAF}}$, I satisfies the formula $\forall y \in Att\ (\ T(y, e_1) \rightarrow (\exists z \in Arg\ (\ S(y, z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg\ (\ S(y, z) \wedge \exists u \in Att\ (\ T(u, z) \wedge \forall v \in Arg\ (\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$. Since I is a model of formulae (4.1), (4.2) and (4.3), then for any $e_2 \in \mathscr{R}$ s.t. $e_1 \in t(e_2)$, there exists $e_3 \in \mathscr{A}$ with $e_3 \in s(e_2)$, and either $I(Unsupportable(e_3)) = \top$, or there exists $e_4 \in \mathscr{R}$ s.t. $e_3 \in t(e_4)$ and for any $e_5 \in \mathscr{A}$ with $e_5 \in s(e_4)$, $e_5 \in S_I$. Since I is a model of axioms (4.11) and (4.12), e_3 and e_5 are unique. Thus for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, there exists $e_3 \in \mathscr{A}$ with $s(e_2) = e_3$, and either $I(Unsupportable(e_3)) = \top$, or there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and $s(e_4) \in S_I$.

- If there exists $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$ and $s(e_4) \in S_I$, then, by Definition 78, $e_3 \in Def(S_I)$. So, since $s(e_2) = e_3$, we have $s(e_2) \in Def(S_I)$.
- If $I(Unsupportable(e_3)) = \top$, by Lemma 29, $e_3 \in UnSupp(S_I)$. So, since $s(e_2) = e_3$, we have $s(e_2) \in UnSupp(S_I)$.

In conclusion, for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, either $s(e_2) \in UnSupp(S_I)$, or $s(e_2) \in Def(S_I)$. By Definition 81, one can deduce that $s(e_2) \in UnAcc(S_I)$ and so, by Definition 82, $e_2 \in UnAct(S_I)$. Thus, $e_1 \in Supp(S_I)$ and for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S_I)$. By Definition 83, this implies that $e_1 \in Acc(S_I)$. □

3. For complete semantics:

⇒ Consider a *complete* extension S of $\mathcal{A}$. A Herbrand interpretation I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ can be defined as follows:

- For any $\dot{e}$ in D_I, $I(Acceptable(e)) = \top$ iff $\forall \alpha \in \mathscr{R}$ s.t. $t(\alpha) = e$, $\alpha \in UnAct(S)$
- The interpretation of the other predicates is defined similarly as in the proof of Proposition 13.2

With this definition, $S_I = S$. It remains to prove that I is a model of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$.

As in the proof of Proposition 13.1, I is a model of formulae (4.11), (4.12) and (4.16).

We must prove that I is a model of $\Sigma_{Rein}(\mathcal{A})$. It is obvious that I is a model of formulae (4.1), (4.2) and (4.3). It remains to show that I is a model of formulae (4.4), (4.5), (4.6), (4.7), (4.8) and (4.9). Since I is a model of formulae (4.16), it is enough to prove that I is a model of formulae (4.4), $(4.5)_{\text{EBAF}}$, $(4.6)_{\text{EBAF}}$, $(4.7)_{\text{EBAF}}$, $(4.8)_{\text{EBAF}}$ and $(4.9)_{\text{EBAF}}$. By definition of I, Formula $(4.5\text{b})_{\text{EBAF}}$ is obviously satisfied. The proofs for the fact that I satisfies formulae $(4.5\text{a})_{\text{EBAF}}$, $(4.6)_{\text{EBAF}}$ and $(4.7)_{\text{EBAF}}$ are similar to those given in the proof of Proposition 13.2.

I satisfies Formula (4.4)**:**

⇒ Let $\dot{e_1} \in D_I$ s.t. $I(Acc(e_1)) = \top$. By the definition of I, $e_1 \in S$. But, S is a *complete* extension, so, by Definition 84, S is *conflict-free* and $S = Acc(S)$; this implies that e_1 *inAcc*(S). By Definition 83, $e_1 \in Supp(S)$, and $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S)$. Consequently, by definition of I, $I(Supported(e_1)) = \top$ and $I(Cand(e_1)) = \top$.

⇐ Let $\dot{e_1} \in D_I$ s.t. $I(Supported(e_1)) = \top$ and $I(Cand(e_1)) = \top$. By definition of I, $e_1 \in Supp(S)$ and $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S)$. By Definition 83, $e_1 \in Acc(S)$. But, S is a *complete* extension, so by Definition 84, S is *conflict-free* and $S = Acc(S)$. This implies that $e_1 \in S$, and so, by definition of I, $I(Acc(e_1)) = \top$.

I **satisfies Formula** $(4.8)_{\text{EBAF}}$**.** Assume that I does not satisfy Formula $(4.8)_{\text{EBAF}}$. So there exists $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$, $I(Cand(e_1)) = \top$ and I does not satisfy the formula $\forall y \in Att(\ T(y, e_1) \rightarrow (\exists z \in Arg(\ S(y,z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y,z) \wedge \exists u \in Att(\ T(u,z) \wedge \forall v \in Arg(\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )$. By definition of I and because I is a model of axioms (4.11) and (4.12), $e_1 \in \mathscr{A}$ and there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, and for any $e_3 \in \mathscr{A}$ with $s(e_2) = e_3$, $e_3 \notin UnSupp(S)$ and for any $e_4 \in \mathscr{R}$ s.t. $t(e_4) = e_3$, there exists $e_5 \in \mathscr{A}$ with $s(e_4) = e_5$ and $e_5 \notin S$. Thus, $e_1 \in \mathscr{A}$ and there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $s(e_2) \notin UnSupp(S)$ and for any $e_4 \in \mathscr{R}$ s.t. $t(e_4) = s(e_2)$, $s(e_4) \notin S$. So, by Definition 78, $s(e_2) \notin Def(S)$. Since $s(e_2) \notin UnSupp(S)$, then, by Definition 81, $s(e_2) \notin UnAcc(S)$ and so, by Definition 82, $e_2 \notin UnAct(S)$. But, $I(Cand(e_1)) = \top$, so, by definition of I, $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S)$. This contradicts the initial assumption.

I **satisfies Formula** $(4.9)_{\text{EBAF}}$**.** Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and I satisfies the formula $\forall y \in Att(\ T(y, e_1) \rightarrow (\exists z \in Arg(\ S(y,z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y,z) \wedge \exists u \in Att(\ T(u,z) \wedge \forall v \in Arg(\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )$. We must show that $I(Cand(e_1)) = \top$. By definition of I, $e_1 \in \mathscr{A}$. Moreover, similarly to the part $\Leftarrow$ of the proof of Proposition 13.2, one can deduce that for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S)$. So, by definition of I, $I(Cand(e_1)) = \top$.

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_d(EBAF) \cup \Sigma_r(\mathcal{A}) \cup (4.16) \cup (4.11) \cup (4.12)$. By the proof of Proposition 13.2, we know that S_I is an *admissible* extension. So, by Definition 84, S_I is *conflict-free* and $S_I \subseteq Acc(S_I)$. It remains to prove that $Acc(S_I) \subseteq S_I$.

Let $e_1 \in \mathscr{A}$ s.t. $e_1 \in Acc(S_I)$. By definition of S_I, we must prove that $I(Acc(e_1)) = \top$. Since I is a model of Formula (4.4), it is enough to show that $I(Supported(e_1)) = \top$ and $I(Cand(e_1)) = \top$. Moreover, since I is a model of formulae (4.1), we know that $I(Arg(e_1)) = \top$.

Proof for $I(Supported(e_1)) = \top$**.** By Definition 18, $e_1 \in Supp(S_I)$, so by Definition 79, $e_1 \in \mathscr{P} \cup \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S_I \cap Supp(S_I \setminus \{t(\alpha)\})\}$. Thus, $e_1 \in \mathscr{P}$ or $e_1 \in \{t(\alpha) \mid \alpha \in \mathscr{S}, s(\alpha) \in S_I \cap Supp(S_I \setminus \{t(\alpha)\})\}$. Consequently, $e_1 \in \mathscr{P}$ or there exists $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$ and $s(e_2) \in S_I$. Therefore, $e_1 \in \mathscr{P}$ or there exists $e_2 \in \mathscr{S}$ s.t. $t(e_2) = e_1$ and there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $e_3 \in S_I$. Since I is a model of formulae (4.1), (4.2) and (4.3), one can deduce that $I(PrimaFacie(e_1)) = \top$ or there exist $\dot{e_2}, \dot{e_3} \in D_I$ with $I(Supp(e_2)) = \top$, $I(Arg(e_3)) = \top$, $I(T(e_2, e_1)) = \top$ and $I(S(e_2, e_3)) = \top$. In addition, by definition of S_I, $I(Acc(e_3)) = \top$. Since I is a model of Axiom (4.11), e_3 is unique, and so $I(PrimaFacie(e_1)) = \top$ or I satisfies the formula $\exists y \in Supp(T(y, e_1) \wedge \forall z \in Arg(S(y,z) \rightarrow Acc(zy)))$. So, I satisfies the formula $PrimaFacie(e_1) \vee \exists y \in Supp(T(y, e_1) \wedge \forall z \in Arg(S(y,z) \rightarrow Acc(z)))$. Since I is a model of Formula $(4.6)_{\text{EBAF}}$, this implies that $I(Supported(e_1)) = \top$.

Proof for $I(Cand(e_1)) = \top$**.** By Definition 18, for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S_I)$. By definitions 82 and 81, for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_2$, there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $e_3 \in Def(S_I)$ or $e_3 \in UnSupp(S_I)$. Two cases are possible.

- If $e_3 \in Def(S_I)$ then, as for the first point of the induction step of Lemma 30, we can show that I satisfies the formula $\forall y \in Att\ (\ T(y, e_1) \rightarrow \exists z \in Arg\ (\ S(y,z) \wedge \exists u \in Att\ (\ T(u,z) \wedge \forall v \in Arg\ (\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )$.
- If $e_3 \in UnSupp(S_I)$, by Lemma 30, $I(Unsupportable(e_3)) = \top$. Thus, for any $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $I(Unsupportable(e_3)) = \top$. Since I is a model of formulae (4.1), (4.2) and (4.3), for any $\dot{e_2} \in D_I$ with $I(Att(e_2)) = \top$ and $I(T(e_2, e_1)) = \top$, there exists $\dot{e_3} \in D_I$ with $I(Arg(e_3)) = \top$, $I(S(e_2, e_3)) = \top$ and $I(Unsupportable(e_3)) = \top$. So, I satisfies the formula $\forall y \in Att\ (\ T(y, e_1) \rightarrow \exists z \in Arg\ (\ S(y,z) \wedge Unsupportable(z)\ )\ )$.

In conclusion, I satisfies the formula $\forall y \in Att\ (\ T(y,e_1) \rightarrow \exists z \in Arg\ (\ S(y,z) \wedge \exists u \in Att\ (\ T(u,z) \wedge \forall v \in Arg\ (\ S(u,v) \rightarrow Acc(v)\ )\ )\ )\ )$ or I satisfies the formula $\forall y \in Att(\ T(y,e_1) \rightarrow \exists z \in Arg(\ S(y,z) \wedge Unsupportable(z)\ )\ )$. So, I satisfies the formula $\forall y \in Att(\ T(y,e_1) \rightarrow (\exists z \in Arg(\ S(y,z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y,z) \wedge \exists u \in Att(\ T(u,z) \wedge \forall v \in Arg(\ S(u,v) \rightarrow Acc(v)\ )\ )\ ))\ )$. Since I is a model of Formula $(4.9)_{\text{EBAF}}$, $I(Cand(e_1)) = \top$. □

4. For the preferred semantics: Let I be an interpretation of a set of formulas Σ. It is obvious to see that I is a $\subseteq$-*maximal* model of Σ iff S_I is $\subseteq$-*maximal* among the extensions S_J, where J is a model of Σ. Considering $\Sigma = \Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$, one can see that the *preferred* extensions correspond to the extensions S_I where I is a $\subseteq$-*maximal* model of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$. □

5. For the grounded semantics: Let I be an interpretation of a set of formulas Σ. It is obvious to see that I is a $\subseteq$-*minimal* model of Σ iff S_I is $\subseteq$-*minimal* among the extensions S_J, where J is a model of Σ. By definition, the *grounded* extension is the *complete* extension that is $\subseteq$-*minimal*. This implies that the *grounded* extension is the extension S_I where I is a $\subseteq$-*minimal* model of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$. □

6. For stable semantics:

   ⇒ Consider a *stable* extension S of $\mathcal{A}$. An interpretation I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$ can be defined as in the proof of Proposition 13.3.

   With this definition, $S_I = S$. It remains to prove that I is a model of formulae $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$.

   As in the proof of Proposition 13.1, we can prove that I is a model of formulae (4.11), (4.12) and (4.16).

   We must prove that I is a model of $\Sigma_{CA}(\mathcal{A})$. It is obvious that I is a model of formulae (4.1), (4.2) and (4.3). It remains to prove that I is a model of formulae (4.4), (4.5), (4.6), (4.7) and (4.10). Since I is a model of formulae (4.16), it is enough to prove that I is a model of formulae (4.4), $(4.5)_{\text{EBAF}}$, $(4.6)_{\text{EBAF}}$, $(4.7)_{\text{EBAF}}$ and $(4.10)_{\text{EBAF}}$. Proving that I satisfies formulae $(4.5)_{\text{EBAF}}$, $(4.6)_{\text{EBAF}}$ and $(4.7)_{\text{EBAF}}$ can be done in the same way as in the proof of Proposition 13.2.

   I satisfies Formula (4.4)**:**

   ⇒ Let $e_1 \in D_I$ s.t. $I(Acc(e_1)) = \top$. By the definition of I, $e_1 \in S$. But, S is a *stable* extension, so, by Definition 84, $S = \mathscr{A} \setminus UnAcc(S)$. Moreover, by Definition 20, S is *conflict-free* and *self-supported*. This implies that $e_1 \in Supp(S)$ and so, by definition of I, $I(Supported(e_1)) = \top$. Assume that $I(Cand(e_1)) = \bot$. By definition of I, this implies that there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$ and $e_2 \notin UnAct(S)$. By Definition 82, $s(e_2) \notin UnAcc$ and so $s(e_2) \in S$. Thus, there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$ and $s(e_2) \in S$, so, by Definition 78, $e_1 \in Def(S)$. As $e_1 \in S$, $S \cap Def(S) \neq \varnothing$; this contradicts the fact that S is *conflict-free*.

   ⇐ Let $e_1 \in D_I$ s.t. $I(Supported(e_1)) = \top$ and $I(Cand(e_1)) = \top$. By definition of I, $e_1 \in Supp(S)$ and $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $e_2 \in UnAct(S)$. Since S is a *stable* extension, $S = \mathscr{A} \setminus UnAcc(S)$. By Definition 82, we also know that $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $s(e_2) \in UnAcc(S)$. So, $\forall e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$, $s(e_2) \notin S$, and thus, by Definition 78, $e_1 \notin Def(S)$. Moreover, $e_1 \in Supp(S)$, so, $e_1 \in Supp(\mathscr{A} \setminus UnAcc(S))$. But, by Definition 81, $Def(S) \subseteq UnAcc(S)$, so, $(\mathscr{A} \setminus UnAcc(S)) \subseteq (\mathscr{A} \setminus Def(S))$ and since $e_1 \in Supp(\mathscr{A} \setminus UnAcc(S))$, by Lemma 15, $e_1 \in Supp(\mathscr{A} \setminus Def(S))$. Thus, $e_1 \notin \mathscr{A} \setminus Supp(\mathscr{A} \setminus Def(S))$ and so, by Definition 80, $e_1 \notin UnSupp(S)$. Since $e_1 \notin Def(S)$ and $x \notin UnSupp(S)$, by Definition 81, $e_1 \notin UnAcc(S)$ and since $S = \mathscr{A} \setminus UnAcc(S)$, $e_1 \in S$. Consequently, by the definition of I, $I(Acc(e_1)) = \top$.

   I satisfies Formula $(4.10a)_{\text{EBAF}}$**:** Let $e_1 \in D_I$ with $I(Arg(e_1)) = \top$ and $I(Cand(e_1)) = \bot$. By definition of I, there exists $e_2 \in \mathscr{R}$ s.t. $t(e_2) = e_1$ and $e_2 \notin UnAct(S)$. By Definition 82, $s(e_2) \notin UnAcc(S)$. Since S is *stable*, by Definition 84, $s(e_2) \in S$. So there exists $e_2 \in \mathscr{R}$ s.t.

$t(e_2) = e_1$ and there exists $e_3 \in \mathscr{A}$ s.t. $e_3 = s(e_2)$ and $e_3 \in S$. Since I is a model of formulae (4.1), (4.2) and (4.3), there exist $\dot{e_2}, \dot{e_3} \in D_I$ with $I(Att(e_2)) = \top$, $I(Arg(e_3)) = \top$, $I(T(e_2, e_1)) = \top$, $I(S(e_2, e_3)) = \top$. Moreover, by definition of I, $I(Acc(e_3)) = \top$. Since I is a model of Axiom (4.11), e_3 is unique, and so I satisfies the formula $\forall z \in Arg(\ S(e_2, z) \rightarrow Acc(z)\ )$. Thus, I satisfies the formula $\exists y \in Arg(\ T(y, e_1) \wedge \forall z \in Arg(\ S(y, z) \rightarrow Acc(z)\ )\ )$. So I is a model of Formula $(4.10a)_{\text{EBAF}}$.

I **satisfies Formula** $(4.10b)_{\text{EBAF}}$**:**. Let $\dot{e_1} \in D_I$ s.t. $I(Arg(e_1)) = \top$ and $I(Supported(e_1)) = \bot$. Since I is a model of Formula $(4.6)_{\text{EBAF}}$, by contraposition, I does not satisfy the formula $PrimaFacie(e_1) \vee \exists y \in Supp\ (\ T(y, e_1) \wedge \forall z \in Arg\ (\ S(y, z) \rightarrow Acc(z)\ )\ )$. So, $I(PrimaFacie(e_1)) = \bot$ and for any $\dot{e_2} \in D_I$ with $I(T(e_2, e_1)) = \bot$, there exists $\dot{e_3} \in D_I$ with $I(Arg(e_3)) = \top$, $I(S(e_2, e_3)) = \top$ and $I(Acc(e_3)) = \bot$. By definition of I, one can deduce that $e_3 \notin S$. Moreover, S is *stable*, so by Definition 84, $e_3 \in UnAcc(S)$. By Definition 81, $e_3 \in Def(S) \cup UnSupp(S)$. So, $e_3 \in Def(S)$ or $e_3 \in UnSupp(S)$. Two cases are possible.

- If $e_3 \in Def(S)$, as in the point "**Proof for** $I(Cand(e_1)) = \top$" given in the proof "For complete semantics", we can show that I satisfies the formula $\exists u \in Att(\ T(u, e_3) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )$. Thus, I satisfies the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$.
- If $a \in UnSupp(S)$, as in the point "**Proof for** $I(Cand(e_1)) = \top$" in the proof "For complete semantics", we can show that $I(Unsupportable(e_3)) = \top$ and so that I satisfies the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ )\ )$.

In conclusion, I satisfies the formula $\forall y \in Supp\ (\ T(y, e_1) \rightarrow \exists z \in Arg\ (\ S(y, z) \wedge \exists u \in Att\ (\ T(u, z) \wedge \forall v \in Arg\ (\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )$ and I satisfies the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow \exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ )\ )$. So, I satisfies the formula $\forall y \in Supp(\ T(y, e_1) \rightarrow (\exists z \in Arg(\ S(y, z) \wedge Unsupportable(z)\ ) \vee \exists z \in Arg(\ S(y, z) \wedge \exists u \in Att(\ T(u, z) \wedge \forall v \in Arg(\ S(u, v) \rightarrow Acc(v)\ )\ )\ )\ )\ )$. Since $I(PrimaFacie(e_1)) = \bot$ and I is a model of Formula $(4.7b)_{\text{EBAF}}$, one can conclude that $I(Unsupportable(e_1)) = \top$. So, I satisfies Formula $(4.10b)_{\text{EBAF}}$.

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.16)\}$. By the proof of Proposition 13.2, we know that S_I is an extension that is *conflict-free* and *self-supported*. So, by the proof of Proposition 21, it remains to show that $(\mathscr{A} \setminus S_I) \subseteq UnAcc(S_I)$.

Let $e_1 \in \mathscr{A} \setminus S_I$. So $e_1 \notin S_I$. By definition of S_I, there exists $\dot{e_1} \in D_1$ with $I(Acc(e_1)) = \bot$. Since I is a model of formulae (4.4), $I(Cand(e_1)) = \bot$ or $I(Supported(e_1)) = \bot$. Two cases are possible.

- If $I(Cand(e_1)) = \bot$, then by Formula $(4.10a)_{\text{EBAF}}$ satisfied by I, we know that I satisfies the formula $\exists y \in Att\ (\ T(y, e_1) \wedge \forall z \in Arg\ (\ S(y, z) \rightarrow Acc(z)\ )\ )$. This implies that there exists $\dot{e_2} \in D_I$ s.t. $I(Att(e_2)) = \top$, $I(T(e_2, e_1)) = \top$ and for any $\dot{e_3} \in D_I$ s.t. $I(Arg(e_3)) = \top$ and $I(S(e_2, e_3)) = \top$, $I(Acc(e_3)) = \top$. Since I is also a model of Axiom (4.11), e_3 is unique. So, there exist $\dot{e_2}, \dot{e_3} \in D_I$ s.t. $I(Att(e_2)) = \top$, $I(Arg(e_3)) = \top$, $I(T(e_2, e_1)) = \top$, $I(S(e_2, e_3)) = \top$ and $I(Acc(e_3)) = \top$. Since I is a model of formulae (4.1), (4.2) and (4.3), there exist $e_2 \in \mathscr{R}$ and $e_3 \in \mathscr{A}$ s.t. $t(e_2) = e_1$ and $s(e_2) = e_3$. Moreover, by definition of S_I, $e_3 \in S_I$, and so $s(e_2) \in S_I$. By Definition 78, $e_1 \in Def(S_I)$. Since $e_1 \in Def(S_I)$, $e_1 \in Def(S_I) \cup UnSupp(S_I)$, and so, by Definition 81, $e_1 \in UnAcc(S_I)$.
- If $I(Supported(e_1)) = \bot$, then by Formula $(4.10b)_{\text{EBAF}}$ satisfied by I, $I(Unsupportable(e_1)) = \top$. By Lemma 29, $e_1 \in UnSupp(S_I)$. Thus, $e_1 \in Def(S_I) \cup UnSupp(S_I)$ and so, by Definition 81, $e_1 \in UnAcc(S_I)$.

□

B.8 Proofs for Section 4.5.7: Theory for HO-EBAF

Proposition. 14 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \mathscr{S}, \mathscr{P}, s, t)$ be an HO-EBAF-C corresponding to an HO-EBAF and $U = (S, \Gamma, \Delta)$ be a structure.*

1. *U is* conflict-free *if and only if there exists a Herbrand model I of $\Sigma_{Coh}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
2. *U is* admissible *if and only if there exists a Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
3. *U is* complete *if and only if there exists a Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
4. *U is* preferred *if and only if there exists a $\subseteq$-*maximal *Herbrand model I of $\Sigma_{Def}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
5. *U is* grounded *if and only if there exists a $\subseteq$-*minimal *Herbrand model I of $\Sigma_{Rein}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*
6. *U is* stable *if and only if there exists a Herbrand model I of $\Sigma_{CA}(\mathcal{A}) \cup \{(4.11), (4.12), (4.17)\}$ such that $S = S_I$, $\Gamma = \Gamma_I$ and $\Delta = \Delta_I$.*

Proof. (for Proposition 14)
As for Proposition 12, the idea is to apply a succession of operations on the language used in this work to retrieve the language that is used in the "Logical Description of a REBAF" and "Logical Formalization of REBAF semantics" sections of [CL20]. In the following, when we mention formulas from [CL20], we refer to formulas given in these particular sections. These operations will then be applied successively to formulas (4.4) and $(4.5)_{\text{HOEBAF}}$ to $(4.10)_{\text{HOEBAF}}$ in order to eventually yield the formulas used in [CL20].

First operation (denoted Θ): correctly rename some of the predicates of the language as follows.

- *Arg* becomes *Arg*
- *Att* becomes *Attack*
- *Sup* becomes *ESupport*
- *PrimaFacie* becomes *PrimaFacie*
- *Supported* becomes *Supp*
- *Unsupportable* becomes *UnSupp*
- *Unacceptable* becomes *NAcc*
- *Cand* becomes $\top$ (*Cand* does not occur in formulas (4.4) and $(4.5)_{\text{HOEBAF}}$ to $(4.10)_{\text{HOEBAF}}$ and has no corresponding predicate in [CL20])
- *Activable* becomes $\top$ (same reason as for *Cand*))
- *Defeated* becomes $\top$ (same reason as for *Cand*)
- *Inhibited* becomes $\top$ (same reason as for *Cand*)
- *Desactivated* becomes $\top$ (same reason as for *Cand*)

Then we separate formulas $\Theta((4.5)_{\text{HOEBAF}})$ to $\Theta((4.10)_{\text{HOEBAF}})$ in two groups of subformulas. In each formula, exactly one quantifier occurs that is bounded to $Arg \cup Attack \cup ESupport$. The separation consists of putting in the first group the formulas with this quantifier bounded to only Arg (group A) and in the second group the formulas with this quantifier bounded to $Attack \cup ESupport$ (group B).

Second operations (denoted Λ): correctly rename the missing unary predicates of the language as follows.

- $Selected(x)$ becomes $eAcc(x)$ when $Arg(x)$ is true
- $Acceptable(x)$ becomes $Acc(x)$ when $Arg(x)$ is true
- $Selected(x)$ becomes $eVal(x)$ when $Attack(x) \vee ESupport(x)$ is true
- $Acceptable(x)$ becomes $Val(x)$ when $Attack(x) \vee ESupport(x)$ is true

Third operation (denoted Π): replace the use of binary predicates S and T by the introduction of functional terms s_α and t_α (justified by the presence of the axioms (4.11) and (4.12) in the theory).

- $\forall a \in Arg(S(\alpha, a) \rightarrow \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $\exists a \in Arg(S(\alpha, a) \wedge \varphi)$ becomes φ in which all occurences of a are replaced by s_α
- $T(\alpha, x)$ becomes $t_\alpha = x$

The idea is then to apply successively Θ, Λ and Π on formulas (4.4) and $(4.5)_{\text{HOEBAF}}$ to $(4.10)_{\text{HOEBAF}}$ so that we obtain the formulas used in [CL20]. However, for some formulas, this result is not immediate. We will therefore use different versions of these formulas (namely (4.4), $(4.5a)_{\text{HOEBAF}}$ and $(4.8)_{\text{HOEBAF}}$) on which to apply Θ, Λ and Π.

Concerning Formula (4.4), we add a boundary on the universal quantifier. This boundary ranges over predicates Arg, Att and Sup. This addition is correct because in Proposition 14 we consider models of theories that contain both (4.4) and (4.2). Thus, instead of (4.4), we consider Formula $(4.4)_{\text{Diff}}$.

$$\forall x \in (Arg \cup Att \cup Sup)(Selected(x) \leftrightarrow (Acceptable(x) \wedge Supported(x))) \qquad ((4.4)_{\text{Diff}})$$

Concerning Formula $(4.5a)_{\text{HOEBAF}}$, the subformula $\exists x \in (Arg \cup Att \cup Sup)\big(T(\alpha, x) \wedge Unacceptable(x)\big)$ is changed into:

$$\forall x \in (Arg \cup Att \cup Sup)\big(T(\alpha, x) \rightarrow Unacceptable(x)\big)$$

This modification is only valid because in Proposition 14 we consider models of theories that contain (4.12). We then use standard modifications that preserve logical equivalence to put the quantifier $\forall x \in (Arg \cup Att \cup Sup)$ at the beginning of the formula. This results in Formula $(4.5a)_{\text{Diff}}$ which will be used instead of $(4.5a)_{\text{HOEBAF}}$.

$$\begin{aligned} &\forall x \in (Arg \cup Att \cup Sup)\Big(\forall \alpha \in Att\Big(\\ &\Big[\forall a \in Arg\big(\ S(\alpha, a) \rightarrow Selected(a)\ \big) \wedge Selected(\alpha) \wedge T(\alpha, x)\Big] \rightarrow \\ &\qquad\qquad Unacceptable(x)\ \Big)\Big) \end{aligned} \qquad ((4.5a)_{\text{Diff}})$$

Concerning Formula $(4.8)_{\text{HOEBAF}}$, we use the same standard modifications that preserve logical equivalence to put the quantifier $\forall \alpha \in Att$ at the beginning of the formula. This results in Formula $(4.8)_{\text{Diff}}$ which will be used instead of $(4.8)_{\text{HOEBAF}}$.

$$\forall\alpha \in Att\Big(\ \forall x \in (Arg \cup Att \cup Sup)\Big(\ [Acceptable(x) \wedge T(\alpha, x)] \rightarrow \Big[\exists a \in Arg\big(\ S(\alpha, a) \wedge Unsupportable(a)\ \big) \vee Unsupportable(\alpha) \vee \exists\beta \in Att\Big(\big[T(\beta, \alpha) \vee \exists a \in Arg(S(\alpha, a) \wedge T(\beta, a))\big] \wedge \big[\forall b \in Arg(\ S(\beta, b) \rightarrow Selected(b)\ ) \wedge Selected(\beta)\big]\Big)\Big]\Big)\Big) \qquad ((4.8)_{\text{Diff}})$$

By applying successively Θ, Λ and Π on formulas $(4.4)_{\text{Diff}}$, $(4.5a)_{\text{Diff}}$, $(4.6)_{\text{HOEBAF}}$, $(4.7)_{\text{HOEBAF}}$, $(4.8)_{\text{Diff}}$, $(4.9)_{\text{HOEBAF}}$, $(4.10)_{\text{HOEBAF}}$, we obtain the following formulas.

$$\forall x \in Arg(eAcc(x) \leftrightarrow (Acc(x) \wedge Supp(x))) \qquad ((4.4)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(eVal(x) \leftrightarrow (Val(x) \wedge Supp(x))) \qquad ((4.4)_{\text{ShiftB}})$$

$$\forall x \in Arg(\forall\alpha \in Attack([eAcc(s_\alpha) \wedge eVal(\alpha) \wedge (t_\alpha = x)] \rightarrow NAcc(x))) \qquad ((4.5a)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(\forall\alpha \in Attack([eAcc(s_\alpha) \wedge eVal(\alpha) \wedge (t_\alpha = x)] \rightarrow NAcc(x))) \qquad ((4.5a)_{\text{ShiftB}})$$

$$\forall x \in Arg(NAcc(x) \rightarrow \neg Acc(x)) \qquad ((4.5b)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(NAcc(x) \rightarrow \neg Val(x)) \qquad ((4.5b)_{\text{ShiftB}})$$

$$\forall x \in Arg([PrimaFacie(x) \vee \exists\alpha \in ESupport((t_\alpha = x) \wedge eAcc(s_\alpha) \wedge eVal(\alpha))] \rightarrow Supp(x)) \qquad ((4.6)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)([PrimaFacie(x) \vee \exists\alpha \in ESupport((t_\alpha = x) \wedge eAcc(s_\alpha) \wedge eVal(\alpha))] \rightarrow Supp(x)) \qquad ((4.6)_{\text{ShiftB}})$$

$$\forall x \in Arg(Supp(x) \rightarrow [PrimaFacie(x) \vee \exists\alpha \in ESupport((t_\alpha = x) \wedge eAcc(s_\alpha) \wedge eVal(\alpha))]) \qquad ((4.7a)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(Supp(x) \rightarrow [PrimaFacie(x) \vee \exists \alpha \in ESupport((t_\alpha = x) \wedge eAcc(s_\alpha) \wedge eVal(\alpha))]) \quad ((4.7a)_{\text{ShiftB}})$$

$$\forall x \in Arg(UnSupp(x) \leftrightarrow [\neg PrimaFacie(x) \wedge \forall \alpha \in ESupport((t_\alpha = x) \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))])]) \quad ((4.7b)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(UnSupp(x) \leftrightarrow [\neg PrimaFacie(x) \wedge \forall \alpha \in ESupport((t_\alpha = x) \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))])]) \quad ((4.7b)_{\text{ShiftB}})$$

$$\forall \alpha \in Attack(\forall x \in Arg([Acc(x) \wedge (t_\alpha = x)] \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))])) \quad ((4.8)_{\text{ShiftA}})$$

$$\forall \alpha \in Attack(\forall x \in (Attack \cup ESupport)([Val(x) \wedge (t_\alpha = x)] \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))])) \quad ((4.8)_{\text{ShiftB}})$$

$$\forall x \in Arg(\forall \alpha \in Attack((t_\alpha = x) \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))]) \rightarrow Acc(x)) \quad ((4.9)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(\forall \alpha \in Attack((t_\alpha = x) \rightarrow [UnSupp(s_\alpha) \vee UnSupp(\alpha) \vee \exists \beta \in Attack([(t_\beta = \alpha) \vee (t_\beta = s_\alpha)] \wedge eAcc(s_\beta) \wedge eVal(\beta))]) \rightarrow Val(x)) \quad ((4.9)_{\text{ShiftB}})$$

$$\forall x \in Arg(\neg Acc(x) \rightarrow \exists \alpha \in Attack(t_\alpha = x \wedge eAcc(s_\alpha) \wedge eVal(\alpha))) \quad ((4.10a)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(\neg Val(x) \rightarrow \exists \alpha \in Attack(t_\alpha = x \wedge eAcc(s_\alpha) \wedge eVal(\alpha))) \qquad ((4.10a)_{\text{ShiftB}})$$

$$\forall x \in Arg(\neg Supp(x) \rightarrow UnSupp(x)) \qquad ((4.10b)_{\text{ShiftA}})$$

$$\forall x \in (Attack \cup ESupport)(\neg Supp(x) \rightarrow UnSupp(x)) \qquad ((4.10b)_{\text{ShiftB}})$$

We have the following immediate results (where "amounts" means "is logically equivalent").

- Formula $(4.4)_{\text{ShiftA}}$ amounts to Formula (2bis) of [CL20]
- Formula $(4.4)_{\text{ShiftB}}$ amounts to Formula (3bis) of [CL20]
- Formula $(4.5a)_{\text{ShiftA}}$ amounts to Formula (2) of [CL20]
- Formulas $(4.5a)_{\text{ShiftB}}$ and $(4.5b)_{\text{ShiftB}}$ together amount to Formula (1) of [CL20]
- Formula $(4.5b)_{\text{ShiftA}}$ amounts to Formula (3) of [CL20]
- Formulas $(4.6)_{\text{ShiftA}}$ and $(4.6)_{\text{ShiftB}}$ together amount to Formula (1bis) of [CL20]
- Formulas $(4.7a)_{\text{ShiftA}}$ and $(4.7a)_{\text{ShiftB}}$ together amount to Formula (17) of [CL20]
- Formulas $(4.7b)_{\text{ShiftA}}$ and $(4.7b)_{\text{ShiftB}}$ together amount to Formula (18) of [CL20]
- Formula $(4.8)_{\text{ShiftA}}$ amounts to Formula (11) of [CL20]
- Formula $(4.8)_{\text{ShiftB}}$ amounts to Formula (12) of [CL20]
- Formula $(4.9)_{\text{ShiftA}}$ amounts to Formula (13) of [CL20]
- Formula $(4.9)_{\text{ShiftB}}$ amounts to Formula (14) of [CL20]
- Formula $(4.10a)_{\text{ShiftA}}$ amounts to Formula (15) of [CL20]
- Formula $(4.10a)_{\text{ShiftB}}$ amounts to Formula (16) of [CL20]
- Formulas $(4.10b)_{\text{ShiftA}}$ and $(4.10b)_{\text{ShiftB}}$ together amount to Formula (19) of [CL20]

The only missing formulas are those that are used to describe the graph of an argumentation framework, namely (4), (4bis), (4ter), (5), (6), (7), (8), (8bis), (8ter), (9) and (10) in [CL20]. They should be retrieved using formulas (4.1), (4.2) and (4.3). Here are the details.

Let us consider formulas $(4.1)_{\text{Shift}}$, $(4.2)_{\text{Shift}}$ and $(4.3)_{\text{Shift}}$, which are the formulas obtained by applying Θ on formulas (4.1), (4.2) and (4.3). We have the following results.

- Formula $(4.2a)_{\text{Shift}}$ amounts to Formula (5) of [CL20]
- Formulas $(4.1a)_{\text{Shift}}$ and $(4.2b)_{\text{Shift}}$ together amount to Formula (7) of [CL20]
- Formulas $(4.1b)_{\text{Shift}}$ and $(4.2c)_{\text{Shift}}$ together amount to Formula (8) of [CL20]
- Formulas $(4.1c)_{\text{Shift}}$ and $(4.2d)_{\text{Shift}}$ together amount to Formula (8bis) of [CL20]
- Formulas $(4.1e)_{\text{Shift}}$ and $(4.1f)_{\text{Shift}}$ together amount to Formula (8ter) of [CL20]
- Formula $(4.1d)_{\text{Shift}}$ amounts to formulas (9) and (10) together of [CL20]

Since in Proposition 14 we consider models of theories that contain (4.11) and (4.12), it is obvious that (4.3) can be rewritten as follows.

$$\text{for all } \alpha \in \mathscr{R} \cup \mathscr{S} \text{ with } s(\alpha) = a \text{ and } t(\alpha) = b,\ \ S(\alpha, a) \wedge T(\alpha, b) \qquad ((4.3)_{\text{bis}})$$

Modifying $(4.3)_{\text{bis}}$ by replacing the predicates S and T by the functional terms s_α and t_α allows us to retrieve Formula (6) of [CL20].

To retrieve fromulas (4), (4bis) and (4ter) of [CL20], we use formulas (4.1a), (4.1b) and (4.1c) of this work. The first issue is that formulas (4.1a), (4.1b) and (4.1c) range over some elements of the argumentation framework, while formulas (4), (4bis) and (4ter) of [CL20] are universal. However, by formulas (4.2) that are satisfied by the models considered in Proposition 14 and the fact that we consider Herbrand models, the sets over which formulas (4.1a), (4.1b) and (4.1c) range form a partition of the model's domain. Thus, we can gather formulas (4.1a), (4.1b) and (4.1c) into a single formula governed by an unbounded universal quantifier, as follows.

$$\begin{aligned} \forall x([&Arg(x) \wedge \neg Att(x) \wedge \neg Sup(x)] \vee \\ [&\neg Arg(x) \wedge Att(x) \wedge \neg Sup(x)] \vee \\ [&\neg Arg(x) \wedge \neg Att(x) \wedge Sup(x)]) \end{aligned} \qquad ((4.1)_{\text{abc}})$$

As formulas (4), (4bis) and (4ter) of [CL20] are all universal formulas, we can also gather them into a single formula, as follows.

$$\begin{aligned} \forall x([&\neg Attack(x) \vee \neg Arg(x)] \wedge \\ [&\neg Attack(x) \vee \neg ESupport(x)] \wedge \\ [&\neg ESupport(x) \vee \neg Arg(x)]) \end{aligned} \qquad (4_{[\text{CL20}]})$$

One can note that Formula $4_{[\text{CL20}]}$ is the closure of a formula in CNF while Formula $(4.1)_{\text{abc}}$ is the closure of a formula in DNF. Let us then compute the CNF of the formula of which $(4.1)_{\text{abc}}$ is the closure. By applying the distributivity property, we obtain the following formula.

$$\begin{aligned} \forall x([&Arg(x) \vee Att(x) \vee Sup(x)] \wedge [\neg Att(x) \vee \neg Arg(x)] \wedge \\ [&\neg Att(x) \vee \neg Arg(x) \vee Sup(x)] \wedge [\neg Att(x) \vee \neg Sup(x) \vee \neg Arg(x)] \wedge \\ [&\neg Att(x) \vee \neg Sup(x)] \wedge [\neg Sup(x) \vee \neg Arg(x)] \wedge \\ [&\neg Sup(x) \vee Att(x) \vee \neg Arg(x)]) \end{aligned}$$

The previous formula can be simplified by removing some conjuncts that contain three terms, because they are subsumed by some other conjuncts. We thus obtain the following formula.

$$\begin{aligned} \forall x([&Arg(x) \vee Att(x) \vee Sup(x)] \wedge \\ [&\neg Att(x) \vee \neg Arg(x)] \wedge [\neg Att(x) \vee \neg Sup(x)] \wedge [\neg Sup(x) \vee \neg Arg(x)]) \end{aligned}$$

If we apply Θ on the previous formula, we obtain formulas (4), (4bis), (4ter) and (5) of [CL20]

In conclusion, since our theory is equivalent to the theory given in [CL20], Proposition 6.1 of [CL20] and Proposition 14 are also equivalent. And so Proposition 14 holds.

□

Appendix C

Proofs of Chapter 5

C.1 Theory for Explanations in Argumentation Frameworks

Theorem. 12 *Let $\mathcal{A} = (\mathscr{A}, \mathscr{R}, \varnothing, \mathscr{A} \cup \mathscr{R}, s, t)$ be an AF and $S \subseteq \mathscr{A}$ be a set of arguments.*

1. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
2. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.5), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
3. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.6), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
4. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.7), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*
5. *$(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ is an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$ if and only if there exists a model I of $\Sigma_2(\mathcal{A}, S)\{(5.8), (4.11), (4.12)\}$ such that $\mathscr{A}' = \mathscr{A}_I$, $\mathscr{R}' = \mathscr{R}_I$, $s' = s_I$ and $t' = t_I$.*

Proof. (for Theorem 12)

1. $\Rightarrow$ Consider an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$ $(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ where $\mathscr{A}' \subseteq \mathscr{A}$, $\mathscr{R}' \subseteq \mathscr{R}$, $s' : \mathscr{R}' \mapsto \mathscr{A}'$ and $t' : \mathscr{R}' \mapsto \mathscr{A}'$. A Herbrand interpretation I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$ can be defined as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \bot$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Expl(e)) = \top$ iff $e \in \mathscr{A}' \cup \mathscr{R}'$
   - For any $\dot{e}$ in D_I, $I(ElemFixed(e)) = \top$ iff $e \in \mathscr{A}$
   - For any $\dot{e}$ in D_I, $I(ElemVar(e)) = \top$ iff $e \in \mathscr{R}$
   - For any $\dot{e}$ in D_I, $I(ExplEF(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(ParticularEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \top$
   - For any $\dot{e}$ in D_I, $I(NecesssaryElemVar(e)) = \bot$

With this definition, $\mathscr{A}_I = \mathscr{A}'$ and $\mathscr{R}_I = \mathscr{R}'$. As such, we have $s_I : \mathscr{R}' \mapsto \mathscr{A}'$ and $t_I : \mathscr{R}' \mapsto \mathscr{A}'$. Moreover, using Definition 77 and this definition of I, we have for $\alpha \in \mathscr{R}'$, $s_I(\alpha) = x$ iff $s(\alpha) = x$ and $t_I(\alpha) = y$ iff $t(\alpha) = y$. Thus, we can deduce that $s_I = s'$ and $t_I = t'$. It remains to prove that I is a model of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$.
I is obviously a model of Axioms (4.1), (4.2), (4.3) and (5.1), and of formulas (5.4). In addition, by definition of I, and because $\mathcal{A}$ is an AF, I is a model of Axioms (4.11) and (4.12).

Consider Formula (5.2a) and let $\dot{e}$ in D_I such that $I(ElemFixed(e)) = \top$. By definition of I, $e \in \mathscr{A}$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{A}$, $e \in \mathscr{A}'$. Then, by Definition 34, $\mathscr{A}' = S$, so $e \in S$. By definition of I, $I(ExplEF(e)) = \top$. The other direction of the equivalence is proved by the reverse reasoning. So, I is a model of Formula (5.2a).

Consider Formula (5.2b) and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $e \in \mathscr{R}$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{R}$, $e \in \mathscr{R}'$. Then, by Definition 34, $s(e) \in S$ and $t(e) \in S$. Let $e_1 = s(e)$ and $e_2 = t(e)$. Since I is a model of Axioms (4.2) and (4.3), there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$ and $I(T(e, e_2)) = \top$. Moreover, as I is a model of Axioms (4.11) and (4.12), $\dot{e_1}$ and $\dot{e_2}$ are unique. As $e_1, e_2 \in S$ by definition of I, we have $I(Selected(e_1)) = I(Selected((e_2)) = \top$. Thus, by definition of I, $I(ParticularEV(e)) = \top$. So, I is a model of Formula (5.2b).

Consider Formula (5.2c) and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $I(NecessaryEV(e)) = \bot$, thus I is a model of Formula (5.2c).

Consider Formula (5.2d), $X = \{\alpha \in \mathscr{R} | s(\alpha) \in S, t(\alpha) \in S\}$ and suppose that there exists $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$, $I(ParticularEV(e)) = \top$ and $I(NecessaryEV(e)) = \bot$. By definition of I, this means that $e \in \mathscr{R}$ and that there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \top$. Still by definition of I, we deduce that $e_1 = s(e)$, $e_2 = t(e)$ and $e_1, e_2 \in S$. Thus, $s(e), t(e) \in S$ and so $e \in X$. As such, $X \neq \varnothing$ so by Definition 34, $\mathscr{R}' \neq \varnothing$. Let $e' \in \mathscr{R}'$. By definition of I and because I is a model of Axioms (4.1) and (4.2), we know that there exists $\dot{e'} \in D_I$ s.t. $I(ElemVar(e')) = \top$, $I(NecessaryEV(e')) = \bot$ and $I(Expl(e')) = \top$. Since $e' \in \mathscr{R}'$, by Definition 34, we have $e' \in X$, so $s(e'), t(e') \in S$. In other terms, there exist $e'_1, e'_2 \in S$ s.t. $e'_1 = s(e')$ and $e'_2 = t(e')$. By definition of I and because I is a model of Axioms (4.1) and (4.2), there thus exist $\dot{e'_1}, \dot{e'_2} \in D_I$ s.t. $I(S(e', e'_1)) = \top$, $I(T(e', e'_2)) = \top$, $I(Selected(e'_1)) = \top$ and $I(Selected(e'_2)) = \top$. By definition of I, we then have $I(ParticularEV(e')) = \top$. In other terms, there exists $\dot{e'} \in D_I$ s.t. $I(ElemVar(e')) = \top$, $I(ParticularEV(e')) = \top$, $I(NecessaryEV(e')) = \bot$ and $I(Expl(e')) = \top$. So I is a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x) \wedge Expl(x))$. And so I is a model of of Formula (5.2d).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_1(\mathcal{A}, S) \cup \{(5.4), (4.11), (4.12)\}$. Consider $X = \{\alpha \in \mathscr{R} | s(\alpha) \in S, t(\alpha) \in S\}$ and suppose that $(\mathscr{A}_I, \mathscr{R}_I, \varnothing, \mathscr{A}_I \cup \mathscr{R}_I, s_I, t_I)$ is not an answer to Q^{Ext}_{Coh} for S on $\mathcal{A}$.

Assume firstly that $\mathscr{A}_I \neq S$. So there exists $e \in \mathscr{A}_I$ s.t. $e \notin S$. As $e \in \mathscr{A}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Arg(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.4), we have $I(ElemFixed(e)) = \top$, and so by Formula (5.2a), $I(ExplEF(e)) = \top$. By using again formulas (5.4), we deduce $I(Selected(e)) = \top$. Finally, as I is a model of Axioms (5.1), we have $e \in S$, a contradiction.

Assume secondly that $\mathscr{R}_I \not\subseteq X$. So there exists $e \in \mathscr{R}_I$ s.t. $s(e) \notin S$ or $t(e) \notin S$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.4), we have

$I(ElemVar(e)) = \top$, and so by Formula (5.2b), $I(ParticularEV(e)) = \top$. By using again formulas (5.4), we deduce that there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \top$. In addition, I is a model of Axiom 4.3, 4.11 and 4.12, so $e_1 = s(e)$ and $e_2 = t(e)$. Finally, as I is a model of Axioms (5.1), we have $e_1 \in S$ and $e_2 \in S$, so $s(e) \in S$ and $t(e) \in S$, a contradiction.

Finally, assume that $X \neq \varnothing$ and that $\mathscr{R}_I = \varnothing$. So there exists $e \in \mathscr{R}$ s.t. $s(e) \in S$ and $t(e) \in S$. Using Axioms (4.1), (4.2), (4.3) and (5.1), there exist $\dot{e}, \dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Att(e)) = \top$, $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \top$. As I is a model of formulas (5.4), we deduce that $I(ElemVar(e)) = \top$, $I(ParticularEV(e)) = \top$ and $I(NecessaryEV(e)) = \bot$. So I is a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x))$. Since I is a model of Formula (5.2d), I is thus also a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x) \wedge Expl(x))$. In other terms, there exists $\dot{e'} \in D_I$ s.t. $I(ElemVar(e')) = \top$, $I(ParticularEV(e')) = \top$, $I(NecessaryEV(e')) = \bot$ and $I(Expl(e')) = \top$. In particular, using formulas (5.4), $I(Att(e')) = \top$. So, using axioms (4.1) and (4.2), we deduce that $e' \in \mathscr{R}$. Moreover, as $I(Expl(e')) = \top$ and $e' \in \mathscr{R}$, by Definition 77, we have $e' \in \mathscr{R}_I$, a contradiction.

2. $\Rightarrow$ Consider an answer to Q_{Def}^{Ext} for S on $\mathcal{A}$ $(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ where $\mathscr{A}' \subseteq \mathscr{A}$, $\mathscr{R}' \subseteq \mathscr{R}$, $s' : \mathscr{R}' \mapsto \mathscr{A}'$ and $t' : \mathscr{R}' \mapsto \mathscr{A}'$. A Herbrand interpretation I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.5), (4.11), (4.12)\}$ can be defined as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \bot$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Expl(e)) = \top$ iff $e \in \mathscr{A}' \cup \mathscr{R}'$
   - For any $\dot{e}$ in D_I, $I(ElemFixed(e)) = \top$ iff $e \in \mathscr{A}$
   - For any $\dot{e}$ in D_I, $I(ElemVar(e)) = \top$ iff $e \in \mathscr{R}$
   - For any $\dot{e}$ in D_I, $I(IsAttacker(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(Att(e_1)) = \top$, $I(S(e_1, e)) = \top$, $I(T(e_1, e_2)) = \top$ and $I(Selected(e_2)) = \top$
   - For any $\dot{e}$ in D_I, $I(ExplEF(e)) = \top$ iff $I(Selected(e)) = \top$ or $I(IsAttacker(e)) = \top$
   - For any $\dot{e}$ in D_I, $I(NecessaryEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(Selected(e_1)) = \top$ and $I(IsAttacker(e_2)) = \top$
   - For any $\dot{e}$ in D_I, $I(AdditionalEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(IsAttacker(e_2)) = \top$
   - For any $\dot{e}$ in D_I, $I(ParticularEF(e)) = \top$ iff $I(IsAttacker(e)) = \top$

   With this definition, $\mathscr{A}_I = \mathscr{A}'$ and $\mathscr{R}_I = \mathscr{R}'$. As such, we have $s_I : \mathscr{R}' \mapsto \mathscr{A}'$ and $t_I : \mathscr{R}' \mapsto \mathscr{A}'$. Moreover, using Definition 77 and this definition of I, we have for $\alpha \in \mathscr{R}'$, $s_I(\alpha) = x$ iff $s(\alpha) = x$ and $t_I(\alpha) = y$ iff $t(\alpha) = y$. Thus, we can deduce that $s_I = s'$ and $t_I = t'$. It remains to prove that I is a model of $\Sigma_2(\mathcal{A}, S) \cup \{(5.5), (4.11), (4.12)\}$.
   I is obviously a model of Axioms (4.1), (4.2), (4.3) and (5.1), and of formulas (5.5). In addition, by definition of I, and because $\mathcal{A}$ is an AF, I is a model of Axioms (4.11) and (4.12).

   Consider Formula (5.3a) and let $\dot{e}$ in D_I such that $I(ElemFixed(e)) = \top$. By definition of I, $e \in \mathscr{A}$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{A}$, $e \in \mathscr{A}'$. Then, by Definition 36, $\mathscr{A}' = S \cup \mathscr{R}^{-1}(S)$. If $e \in S$, by definition of I, $I(Selected(e)) = \top$. Otherwise, $e \in \mathscr{R}^{-1}(S)$, which means there exists $e_1 \in \mathscr{R}$ s.t. $s(e_1) = e$ and $t(e_1) = e_2$ with $e_2 \in S$. Since I is a model of Axioms (4.1), (4.2) and (4.3), there exist $\dot{e_1}, \dot{e_2}$ in D_I such that

$I(Att(e_1)) = \top$, $I(T(e_1, e)) = \top$, $I(S(e_1, e_2)) = \top$. Moreover, as I is a model of Axiom (5.1), $I(Selected(e_2)) = \top$. Thus, by definition of I, $I(IsAttacker(e)) = \top$. As such, I is a model of the formula $Selected(e) \vee IsAttacker(e)$. Using Formulas (5.5) then gives $I(ExplEF(e)) = \top$. The other direction of the equivalence is proved by the reverse reasoning. So, I is a model of Formula (5.3a).

Consider Formula (5.3b), $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(S), t(\alpha) \in S\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $e \in \mathscr{R}$. Suppose that $I(NecessaryEV(e)) = \top$. According to Formulas (5.5), this means that there exist $\dot{e_1}, \dot{e_2}$ in D_I such that $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(Selected(e_1)) = \top$ and $I(IsAttacker(e_2)) = \top$. Since I is a model of Axioms (4.2), (4.3), (4.11) and (4.12), this means that $e_2 = s(e)$ and $e_1 = t(e)$. In addition, by definition of I, $e_1 \in S$. With a similar reasoning as in the previous point, from $I(IsAttacker(e_2)) = \top$, we deduce that $e_2 \in \mathscr{R}^{-1}(S)$. As such, $e \in X$, so by Definition 36, $e \in \mathscr{R}'$. By definition of I we thus have $I(Expl(e)) = \top$. So, I is a model of Formula (5.3b).

Consider Formula (5.3c), $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(S), t(\alpha) \in S\}$, $Y = \{\alpha \in \mathscr{R} \mid s(\alpha) \in S, t(\alpha) \in \mathscr{R}^{-1}(S)\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{R}$, $e \in \mathscr{R}'$. By Definition 36, we thus have $e \in X \cup Y$. If $e \in X$, with a similar reasoning as in the previous point, we deduce that $I(NecessaryEV(e)) = \top$. If $e \in Y$, since I is a model of Axioms (4.2), (4.3), (4.11) and (4.12), there exist $\dot{e_1}, \dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$ and $I(Selected(e_1)) = \top$. In addition, as $e_2 = t(e)$ and $t(e) \in \mathscr{R}^{-1}(S)$, with a similar reasoning as in the proof for Formula (5.3a), we deduce that $I(IsAttacker(e_2)) = \top$. Thus, using Formulas (5.5), we have $I(AdditionalEV(e)) = \top$. As such, I is a model of the formula $NecessaryEV(e) \vee AdditionalEV(e)$. So, I is a model of Formula (5.3c).

Consider Formula (5.3d), let $\dot{e}$ in D_I such that $I(ParticularEF(e)) = \top$. By definition of I, this means that $I(IsAttacker(e)) = \top$. With a similar reasoning as in the proof for Formula (5.3a), we deduce that $e \in \mathscr{R}^{-1}(S)$. Moreover suppose that there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Att(e_1)) = \top$, $I(Selected(e_2)) = \top$, $I(T(e_1, e)) = \top$ and $I(S(e_1, e_2)) = \top$. In other terms, I is a model of the formula $\exists \alpha, a(Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. Still by definition of I, we also deduce that $e = t(e_1)$ and $e_2 = s(e_1)$. Thus, $s(e_1) \in S$, so $e \in \mathscr{R}^{+1}(S)$. As such, by Definition 36, there exists $e'_1 \in \mathscr{R}'$ s.t. $t(e'_1) = e$ and $s(e'_1) = e'_2$ with $e'_2 \in S$. By definition of I and because I is a model of Axioms (4.1), (4.3) and (4.2), we thus know that there exist $\dot{e'_1}, \dot{e'_2} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(Expl(e'_1)) = \top$, $I(Selected(e'_2)) = \top$, $I(T(e'_1, e)) = \top$ and $I(S(e'_1, e'_2)) = \top$. So I is a model of the formula $\exists \beta, b(Att(\beta) \wedge T(\beta, e) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta))$. And so I is a model of of Formula (5.3d).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_2(\mathcal{A}, S) \cup \{(5.5), (4.11), (4.12)\}$. Consider $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(S), t(\alpha) \in S\}$, $Y = \{\alpha \in \mathscr{R} \mid s(\alpha) \in S, t(\alpha) \in \mathscr{R}^{-1}(S)\}$ and suppose that $(\mathscr{A}_I, \mathscr{R}_I, \varnothing, \mathscr{A}_I \cup \mathscr{R}_I, s_I, t_I)$ is not an answer to Q^{Ext}_{Def} for S on $\mathcal{A}$.

Assume firstly that $\mathscr{A}_I \neq S \cup \mathscr{R}^{-1}(S)$. Thus, either $\mathscr{A}_I \not\subseteq S \cup \mathscr{R}^{-1}(S)$ or $S \cup \mathscr{R}^{-1}(S) \not\subseteq \mathscr{A}_I$. Let $e \in \mathscr{A}_I$ and suppose $e \notin S \cup \mathscr{R}^{-1}(S)$. So, $e \notin S$ and $e \notin \mathscr{R}^{-1}(S)$. By Definition 77, and because I is a model of Axioms (4.1) and (4.2), there exists $\dot{e} \in D_I$ s.t. $I(Arg(e)) = \top$ and $I(Expl(e)) = \top$. Using Formulas (5.5), we thus have $I(ElemFixed(e)) = \top$. Then, Formula (5.3a) gives us $I(ExplEF(e)) = \top$, which, using again Formulas (5.5), results in $I(Selected(e)) = \top$ or $I(IsAttacker(e)) = \top$. If $I(Selected(e)) = \top$, because I is a model of Axioms (5.1), we deduce $e \in S$, a contradiction. If $I(IsAttacker(e)) = \top$, with a similar reasoning as in the previous points, we deduce that $e \in \mathscr{R}^{-1}(S)$, another contradiction. The case where $S \cup \mathscr{R}^{-1}(S) \not\subseteq \mathscr{A}_I$ is dealt with using the reverse reasoning.

Assume secondly that $X \not\subseteq \mathscr{R}_I$. So there exists $e \in \mathscr{R}$ s.t. $s(e) \in \mathscr{R}^{-1}(S)$, $t(e) \in S$, and $e \notin \mathscr{R}_I$. As $e \in \mathscr{R}$ and $e \notin \mathscr{R}_I$, and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \bot$. Thus, using Formulas (5.5) we deduce $I(ElemVar(e)) = \top$. Moreover, as I is a model of Axioms (4.3), (4.11) and (4.12), there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_2)) = \top$ and $I(T(e, e_1)) = \top$. Using Axiom (5.1) we get $I(Selected(e_1)) = \top$ and with a similar reasoning as in the previous points, from $s(e) \in \mathscr{R}^{-1}(S)$ and $s(e) = e_2$, we get $I(IsAttacker(e_2)) = \top$. Then, Formulas (5.5) yield $I(NecessaryEV(e)) = \top$ and thus, by Formula (5.3b) we obtain $I(Expl(e)) = \top$, a contradiction.

Assume thirdly that $\mathscr{R}_I \not\subseteq X \cup Y$. So there exists $e \in \mathscr{R}_I$ s.t. $e \notin X$ and $e \notin Y$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.5), we have $I(ElemVar(e)) = \top$, so by Formula (5.3c), $I(NecessaryEV(e)) = \top$ or $I(AdditionalEV(e)) = \top$. If $I(NecessaryEV(e)) = \top$, by formulas (5.5) there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(Selected(e_1)) = \top$ and $I(IsAttacker(e_2)) = \top$. Because I is a model of Axioms (4.3), (4.2), (4.11) and (4.12), we have $s(e) = e_2$ and $t(e) = e_1$. In addition, by Axiom (5.1) $t(e) \in S$ and with a similar reasoning as in the previous points, with $I(IsAttacker(e_2)) = \top$ and $I(S(e, e_2)) = \top$ we deduce that $s(e) \in \mathscr{R}^{-1}(S)$. As such, we have $e \in X$, a contradiction. In the case where $I(AdditionalEV(e)) = \top$, with a similar reasoning, we deduce that $e \in Y$, a contradiction as well.

Finally, assume that there exists $e \in \mathscr{R}^{-1}(S)$ with $e \in \mathscr{R}^{+1}(S)$ and s.t. there exists no $\alpha \in \mathscr{R}_I$ with $t(\alpha) = e$ and $s(\alpha) \in S$. Using Axioms (4.1), (4.2), (4.3) and (5.1), there exist $\dot{e}, \dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Arg(e)) = \top$, $I(Att(e_1)) = \top$, $I(Arg(e_2)) = \top$, $I(T(e_1, e)) = \top$, $I(S(e_1, e_2)) = \top$, $I(Selected(e_2)) = \top$. In addition, since $e \in \mathscr{R}^{-1}(S)$, with a similar reasoning as in the previous points, we deduce that $I(IsAttacker(e)) = \top$. As I is a model of formulas (5.5), we deduce that $I(ParticularEF(e)) = \top$. In addition, I is a model of the formula $\exists \alpha, a\ (Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. Since I is a model of Formula (5.3d), I is thus also a model of the formula $\exists \beta, b\ (Att(\beta) \wedge T(\beta, e) \wedge S(\beta, a) \wedge Selected(a) \wedge Expl(\beta))$. In other terms, there exist $\dot{e'_1}, \dot{e'_2} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(T(e'_1, e)) = \top$, $I(S(e'_1, e'_2)) = \top$, $I(Selected(e'_2)) = \top$ and $I(Expl(e'_1)) = \top$. So, using axioms (4.1), (4.3), (4.2) and (5.1), and Definition 77 we deduce that there exists $e'_1 \in \mathscr{R}_I$ s.t. $t(e'_1) = e$, $s(e'_1) = e'_2$ and $e'_2 \in S$, a contradiction.

3. $\Rightarrow$ Consider an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$ $(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ where $\mathscr{A}' \subseteq \mathscr{A}$, $\mathscr{R}' \subseteq \mathscr{R}$, $s' : \mathscr{R}' \mapsto \mathscr{A}'$ and $t' : \mathscr{R}' \mapsto \mathscr{A}'$. A Herbrand interpretation I of $\Sigma_1(\mathcal{A}, S) \cup \{(5.6), (4.11), (4.12)\}$ can be defined as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \bot$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Expl(e)) = \top$ iff $e \in \mathscr{A}' \cup \mathscr{R}'$
   - For any $\dot{e}$ in D_I, $I(ElemFixed(e)) = \top$ iff $e \in \mathscr{R}$
   - For any $\dot{e}$ in D_I, $I(ElemVar(e)) = \top$ iff $e \in \mathscr{A}$
   - For any $\dot{e}$ in D_I, $I(ExplEF(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(ParticularEV(e)) = \top$ iff for all $\dot{e'}$ in D_I such that $I(Att(e')) = \top$, we have $I(T(e', e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(NecessaryEV(e)) = \top$ iff $I(Selected(e)) = \top$

   With this definition, $\mathscr{A}_I = \mathscr{A}'$ and $\mathscr{R}_I = \mathscr{R}'$. As such, we have $s_I : \mathscr{R}' \mapsto \mathscr{A}'$ and $t_I : \mathscr{R}' \mapsto \mathscr{A}'$. Moreover, using Definition 77 and this definition of I, we have for $\alpha \in \mathscr{R}'$, $s_I(\alpha) = x$ iff $s(\alpha) = x$

and $t_I(\alpha) = y$ iff $t(\alpha) = y$. Thus, we can deduce that $s_I = s'$ and $t_I = t'$. It remains to prove that I is a model of $\Sigma_1(\mathcal{A}, S) \cup \{(5.6), (4.11), (4.12)\}$.

I is obviously a model of Axioms (4.1), (4.2), (4.3) and (5.1), and of formulas (5.6). In addition, by definition of I, and because $\mathcal{A}$ is an AF, I is a model of Axioms (4.11) and (4.12).

Consider Formula (5.2a) and let $\dot{e}$ in D_I such that $I(ElemFixed(e)) = \top$. By definition of I, $e \in \mathscr{R}$. Moreover, still by definition of I, $I(ExplEF(e)) = \bot$. Thus, we must prove that $I(Expl(e)) = \bot$. This equates to proving that $e \notin \mathscr{A}'$ and $e \notin \mathscr{R}'$. Since $e \in \mathscr{R}$ and $\mathscr{A}' \subseteq \mathscr{A}$, we obviously have $e \notin \mathscr{A}'$. In addition, by Definition 38, $\mathscr{R}' = \varnothing$, ensuring that $e \notin \mathscr{R}'$. So, I is a model of Formula (5.2a).

Consider Formula (5.2b), $X = \{a \in \mathscr{A} | \mathscr{R}^{-1}(a) = \varnothing\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $e \in \mathscr{A}$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{A}$, $e \in \mathscr{A}'$. Then, by Definition 38, $e \in X$ and so, $\mathscr{R}^{-1}(e) = \varnothing$. In other terms, there exists no $e' \in \mathscr{R}$ such that $t(e') = e$. Since I is a model of Axioms (4.1), (4.3) and (4.2), there exists no $\dot{e'} \in D_I$ such that $I(Att(e')) = \top$ and $I(T(e', e)) = \top$. Equivalently, for all $\dot{e'} \in D_I$ such that $I(Att(e')) = \top$, $I(T(e', e)) = \bot$. Thus, by definition of I, $I(ParticularEV(e)) = \top$. So, I is a model of Formula (5.2b).

Consider Formula (5.2c), $X = \{a \in \mathscr{A} | \mathscr{R}^{-1}(a) = \varnothing\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. Suppose that $I(ParticularEV(e)) = \top$ and $I(NecessaryEV(e)) = \top$. By definition of I, this firstly mean that $e \in \mathscr{A}$ and $I(Selected(e)) = \top$, and so that $e \in S$. Secondly, this also means that for all $\dot{e'} \in D_I$ such that $I(Att(e')) = \top$, $I(T(e', e)) = \bot$. Since I is a model of Axioms (4.1), (4.3) and (4.2) and by definition of I, this means that there are no $e' \in \mathscr{R}$ such that $t(e') = \varnothing$. Thus, $\mathscr{R}^{-1}(e) = \varnothing$, and so $e \in X$. As $e \in S$ and $e \in X$, $e \in S \cap X$, which, by definition 38, means that $e \in \mathscr{A}'$. By definition of I, we deduce that $I(Expl(e)) = \top$. So, I is a model of Formula (5.2c).

Consider Formula (5.2d), $X = \{a \in \mathscr{A} | \mathscr{R}^{-1}(a) = \varnothing\}$ and suppose that there exists $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$, $I(ParticularEV(e)) = \top$ and $I(NecessaryEV(e)) = \bot$. By definition of I, this means that $e \in \mathscr{A}$, for all $\dot{e_1} \in D_I$ such that $I(Att(e_1)) = \top$, $I(T(e_1, e)) = \bot$, and $I(Selected(e)) = \bot$. Thus, still by definition of I, $e \notin S$. Since I is a model of Axioms (4.1), (4.3) and (4.2) and by definition of I, this means that there are no $e_1 \in \mathscr{R}$ such that $t(e_1) = e$. Thus, $\mathscr{R}^{-1}(e) = \varnothing$, and so $e \in X$. Hence, $e \in (\mathscr{A} \setminus S) \cap X$. By definition 38, we deduce that there exists $e' \in (\mathscr{A} \setminus S) \cap X$ with $e' \in \mathscr{A}'$. By definition of I and because I is a model of Axioms (4.1) and (4.2), we know that there exists $\dot{e'} \in D_I$ such that $I(Arg(e')) = \top$, $I(Selected(e')) = \bot$ and $I(Expl(e')) = \top$. In particular, by definition of I, we also have $I(ElemVar(e')) = \top$ and $I(NecessaryEV(e')) = \bot$. As $e' \in X$, there are no $e'_1 \in \mathscr{R}$ such that $t(e'_1) = e'$. Since I is a model of Axioms (4.1), (4.3) and (4.2), there exists no $\dot{e'_1} \in D_I$ such that $I(Att(e'_1)) = \top$ and $I(T(e'_1, e')) = \top$. Thus, by definition of I, $I(ParticularEV(e')) = \top$. As such, $I(ElemVar(e')) = \top$, $I(ParticularEV(e')) = \top$, $I(NecessaryEV(e')) = \bot$ and $I(Expl(e')) = \top$. So I is a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x) \wedge Expl(x))$. And so I is a model of of Formula (5.2d).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_1(\mathcal{A}, S) \cup \{(5.6), (4.11), (4.12)\}$. Consider $X = \{a \in \mathscr{A} | \mathscr{R}^{-1}(a) = \varnothing\}$ and suppose that $(\mathscr{A}_I, \mathscr{R}_I, \varnothing, \mathscr{A}_I \cup \mathscr{R}_I, s_I, t_I)$ is not an answer to Q^{Ext}_{Rein1} for S on $\mathcal{A}$.

Assume firstly that $\mathscr{R}_I \neq \varnothing$. So there exists $e \in \mathscr{R}_I$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.6), we have $I(ElemFixed(e)) = \top$, and $I(ExplEF(e)) = \bot$. Finally, by Formula (5.2a), $I(Expl(e)) = \bot$, a contradiction.

Assume secondly that $\mathscr{A}_I \not\subseteq X$. So there exists $e \in \mathscr{A}_I$ s.t. $\mathscr{R}^{-1}(e) \neq \varnothing$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Arg(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.6), we have $I(ElemVar(e)) = \top$, and so by Formula (5.2b), $I(ParticularEV(e)) = \top$. By using again formulas (5.6), we deduce that for all $\dot{e'} \in D_I$ s.t. $I(Att(e')) = \top$, $I(T(e', e)) = \bot$. In addition, I is a model of Axiom (4.3) and 4.12, so for all $e' \in \mathscr{R}$, $t(e') \neq e$. Finally, this means that $\mathscr{R}^{-1}(e) = \varnothing$, a contradiction.

Assume thirdly that $S \cap X \not\subseteq \mathscr{A}_I$. So, there exists e s.t. $e \in S$, $e \in X$ and $e \notin \mathscr{A}_I$. By Definition 77, either $e \notin \mathscr{A}$ or $I(Expl(e)) = \bot$. By definition, $S \subseteq \mathscr{A}$, so $e \in \mathscr{A}$, thus it must be the case that $I(Expl(e)) = \bot$. As $e \in S$ and $e \in X$, and because I is a model of Axioms (5.1), (4.1), (4.2) and (4.12), we have $I(Selected(e)) = \top$, $I(Arg(e)) = \top$ and there exists no $\dot{e'} \in D_I$ s.t. $I(Att(e')) = \top$ and $I(T(e', e)) = \top$. In other terms, for all $\dot{e'} \in D_I$ s.t. $I(Att(e')) = \top$, $I(T(e', e)) = \bot$. Using Formulas (5.6), we deduce that $I(ElemVar(e)) = \top$, $I(NecessaryEV(e)) = \top$ and $I(ParticularEV(e)) = \top$. Then, using Formula (5.2c) we obtain $I(Expl(e)) = \top$. As $e \in \mathscr{A}$ and $I(Expl(e)) = \top$, by Definition 77 we have $e \in \mathscr{A}_I$, a contradiction.

Finally, assume that $(A \setminus S) \cap X \neq \varnothing$ and that there exists no $a \in (A \setminus S) \cap X$ s.t. $a \in \mathscr{A}_I$. So there exists e s.t. $e \in \mathscr{A}$, $e \in X$ and $e \notin S$. Using Axioms (4.1), (4.2), (4.3) and (5.1), there exists $\dot{e} \in D_I$ s.t. $I(Arg(e)) = \top$, $I(Selected(e)) = \bot$, and for all $\dot{e_1} \in D_I$ s.t. $I(Att(e_1)) = \top$, $I(S(e_1, e)) = \bot$. As I is a model of formulas (5.6), we deduce that $I(ElemVar(e)) = \top$, $I(ParticularEV(e)) = \top$ and $I(NecessaryEV(e)) = \bot$. So I is a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x))$. Since I is a model of Formula (5.2d), I is thus also a model of the formula $\exists x \in ElemVar\ (ParticularEV(x) \wedge \neg NecessaryEV(x) \wedge Expl(x))$. In other terms, there exists $\dot{e'} \in D_I$ s.t. $I(ElemVar(e')) = \top$, $I(ParticularEV(e')) = \top$, $I(NecessaryEV(e')) = \bot$ and $I(Expl(e')) = \top$. Using again formulas (5.6), $I(Arg(e')) = \top$, $I(Selected(e')) = \bot$ and for all $\dot{e'_1} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(T(e'_1, e')) = \bot$. Thus, by Axioms (4.1), (4.2), (4.3) and (4.12), $e' \in \mathscr{A}$, $e' \notin S$ and $e' \in X$. Moreover, as $I(Expl(e')) = \top$ and $e' \in \mathscr{A}$, by Definition 77, we have $e' \in \mathscr{A}_I$, a contradiction.

4. $\Rightarrow$ Consider an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$ $(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ where $\mathscr{A}' \subseteq \mathscr{A}$, $\mathscr{R}' \subseteq \mathscr{R}$, $s' : \mathscr{R}' \mapsto \mathscr{A}'$ and $t' : \mathscr{R}' \mapsto \mathscr{A}'$. A Herbrand interpretation I of $\Sigma_2(\mathcal{A}, S) \cup \{(5.7), (4.11), (4.12)\}$ can be defined as follows:
   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \bot$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Expl(e)) = \top$ iff $e \in \mathscr{A}' \cup \mathscr{R}'$
   - For any $\dot{e}$ in D_I, $I(ElemFixed(e)) = \top$ iff $e \in \mathscr{A}$
   - For any $\dot{e}$ in D_I, $I(ElemVar(e)) = \top$ iff $e \in \mathscr{R}$
   - For any $\dot{e}$ in D_I, $I(IsDefended(e)) = \top$ iff there exist $\dot{e_1}$, $\dot{e_2}$ $\dot{e_3}$ and $\dot{e_4}$ in D_I such that $I(Att(e_1)) = \top$, $I(Att(e_2)) = \top$, $I(T(e_2, e)) = \top$, $I(S(e_2, e_4)) = \top$, $I(T(e_1, e_4)) = \top$, $I(S(e_1, e_3)) = \top$ and $I(Selected(e_3)) = \top$
   - For any $\dot{e}$ in D_I, $I(IsAttackerOfDefended(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(Att(e_1)) = \top$, $I(S(e_1, e)) = \top$, $I(T(e_1, e_2)) = \top$ and $I(IsDefended(e_2)) = \top$
   - For any $\dot{e}$ in D_I, $I(ExplEF(e)) = \top$ iff $I(Selected(e)) = \top$ or $I(IsDefended(e)) = \top$ or $I(IsAttackerOfDefended(e)) = \top$
   - For any $\dot{e}$ in D_I, $I(NecessaryEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(IsDefended(e_1)) = \top$ and $I(IsAttackerOfDefended(e_2)) = \top$

- For any $\dot{e}$ in D_I, $I(AdditionalEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(IsAttackerOfDefended(e_2)) = \top$
- For any $\dot{e}$ in D_I, $I(ParticularEF(e)) = \top$ iff $I(IsAttackerOfDefended(e)) = \top$

With this definition, $\mathscr{A}_I = \mathscr{A}'$ and $\mathscr{R}_I = \mathscr{R}'$. As such, we have $s_I : \mathscr{R}' \mapsto \mathscr{A}'$ and $t_I : \mathscr{R}' \mapsto \mathscr{A}'$. Moreover, using Definition 77 and this definition of I, we have for $\alpha \in \mathscr{R}'$, $s_I(\alpha) = x$ iff $s(\alpha) = x$ and $t_I(\alpha) = y$ iff $t(\alpha) = y$. Thus, we can deduce that $s_I = s'$ and $t_I = t'$. It remains to prove that I is a model of $\Sigma_2(\mathcal{A}, S) \cup \{(5.7), (4.11), (4.12)\}$.

I is obviously a model of Axioms (4.1), (4.2), (4.3) and (5.1), and of formulas (5.7). In addition, by definition of I, and because $\mathcal{A}$ is an AF, I is a model of Axioms (4.11) and (4.12).

Consider Formula (5.3a) and let $\dot{e}$ in D_I such that $I(ElemFixed(e)) = \top$. By definition of I, $e \in \mathscr{A}$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{A}$, $e \in \mathscr{A}'$. Then, by Definition 39, $\mathscr{A}' = S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. In the first case, if $e \in S$, by definition of I, $I(Selected(e)) = \top$. In the second case, if $e \in \mathscr{R}^{+2}(S)$, it means that there exist $e_1, e_2 \in \mathscr{R}$ s.t. $t(e_2) = e$, $s(e_2) = e_4$, $t(e_1) = e_4$, $s(e_2) = e_3$ and $e_3 \in S$. Since I is a model of Axioms (4.1), (4.2) and (4.3), there exist $\dot{e_1}, \dot{e_2}, \dot{e_3}, \dot{e_4}$ in D_I such that $I(Att(e_1)) = \top$, $I(Att(e_2)) = \top$, $I(T(e_2, e)) = \top$, $I(S(e_2, e_4)) = \top$, $I(T(e_1, e_4)) = \top$, $I(S(e_1, e_3)) = \top$. Moreover, as I is a model of Axiom (5.1), $I(Selected(e_3)) = \top$. Thus, by definition of I, $I(IsDefended(e)) = \top$. In the third case, if $e \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, it means that there exists $e_1 \in \mathscr{R}$ s.t. $s(e_1) = e, t(e_1) = e_2, e_2 \in \mathscr{R}^{+2}(S)$. Since I is a model of Axioms (4.1), (4.2) and (4.3), there exist $\dot{e_1}, \dot{e_2}$ in D_I such that $I(Att(e_1)) = \top$, $I(S(e_1, e)) = \top$, $I(T(e_1, e_2)) = \top$. Moreover, with a similar reasoning as in the previous case, $I(IsDefended(e_2)) = \top$. Thus, by definition of I, $I(IsAttackerOfDefended(e)) = \top$. As such, I is a model of the formula $Selected(e) \vee IsAttacker(e) \vee IsAttackerOfDefended(e)$. Using Formulas (5.7) then gives $I(ExplEF(e)) = \top$. The other direction of the equivalence is proved by the reverse reasoning. So, I is a model of Formula (5.3a).

Consider Formula (5.3b), $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), t(\alpha) \in \mathscr{R}^{+2}(S)\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $e \in \mathscr{R}$. Suppose that $I(NecessaryEV(e)) = \top$. According to Formulas (5.7), this means that there exist $\dot{e_1}, \dot{e_2}$ in D_I such that $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(IsDefended(e_1)) = \top$ and $I(IsAttackerOfDefended(e_2)) = \top$. Since I is a model of Axioms (4.2), (4.3), (4.11) and (4.12), this means that $e_2 = s(e)$ and $e_1 = t(e)$. In addition, with a similar reasoning as in the previous point, from $I(IsDefended(e_1)) = \top$ and $I(IsAttackerOfDefended(e_2)) = \top$, we deduce that $e_1 \in \mathscr{R}^{+2}(S)$ and $e_2 \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. As such, $e \in X$, so by Definition 39, $e \in \mathscr{R}'$. By definition of I we thus have $I(Expl(e)) = \top$. So, I is a model of Formula (5.3b).

Consider Formula (5.3c), $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), t(\alpha) \in \mathscr{R}^{+2}(S)\}$, $Y = \{\alpha \in \mathscr{R} \mid s(\alpha) \in S, t(\alpha) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$ and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and $e \in \mathscr{R}$, so $e \in \mathscr{R}'$. By Definition 39, we thus have $e \in X \cup Y$. If $e \in X$, with a similar reasoning as in the previous point, we deduce that $I(NecessaryEV(e)) = \top$. If $e \in Y$, since I is a model of Axioms (4.2), (4.3), (4.11) and (4.12), there exist $\dot{e_1}, \dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$ and $I(Selected(e_1)) = \top$. In addition, as $e_2 = t(e)$ and $t(e) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, with a similar reasoning as in the proof for Formula (5.3a), we deduce that $I(IsAttackerOfDefended(e_2)) = \top$. Thus, using Formulas (5.7), we have $I(AdditionalEV(e)) = \top$. As such, I is a model of the formula $NecessaryEV(e) \vee AdditionalEV(e)$. So, I is a model of Formula (5.3c).

Consider Formula (5.3d), let $\dot{e}$ in D_I such that $I(ParticularEF(e)) = \top$ and suppose that there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Att(e_1)) = \top$, $I(Selected(e_2)) = \top$, $I(T(e_1, e)) = \top$ and $I(S(e_1, e_2)) = \top$. In other terms, I is a model of the formula $\exists \alpha, a(Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. By

definition of I, $I(ParticularEF(e)) = \top$ means that $I(IsAttackerOfDefended(e)) = \top$. With a similar reasoning as in the proof for Formula (5.3a), we deduce that $e \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. Still by definition of I, we also deduce that $e = t(e_1)$ and $e_2 = s(e_1)$. Thus, $s(e_1) \in S$, so $e \in \mathscr{R}^{+1}(S)$. As such, by Definition 39, there exists $e'_1 \in \mathscr{R}'$ s.t. $t(e'_1) = e$ and $s(e'_1) = e'_2$ with $e'_2 \in S$. By definition of I and because I is a model of Axioms (4.1), (4.3) and (4.2), we thus know that there exist $\dot{e'_1}\dot{e'_2} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(Expl(e'_1)) = \top$, $I(Selected(e'_2)) = \top$, $I(T(e'_1, e)) = \top$ and $I(S(e'_1, e'_2)) = \top$. So I is a model of the formula $\exists \beta, b(Att(\beta) \wedge T(\beta, e) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta))$. And so I is a model of of Formula (5.3d).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_2(\mathcal{A}, S) \cup \{(5.7), (4.11), (4.12)\}$. Consider $X = \{\alpha \in \mathscr{R} \mid s(\alpha) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)), t(\alpha) \in \mathscr{R}^{+2}(S)\}$, $Y = \{\alpha \in \mathscr{R} \mid s(\alpha) \in S, t(\alpha) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))\}$ and suppose that $(\mathscr{A}_I, \mathscr{R}_I, \varnothing, \mathscr{A}_I \cup \mathscr{R}_I, s_I, t_I)$ is not an answer to Q^{Ext}_{Rein2} for S on $\mathcal{A}$.

Assume firstly that $\mathscr{A}_I \neq S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. Thus, either $\mathscr{A}_I \not\subseteq S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ or $S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \not\subseteq \mathscr{A}_I$. Let $e \in \mathscr{A}_I$ and suppose $e \notin S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. So, $e \notin S$ and $e \notin \mathscr{R}^{+2}(S)$ and $e \notin \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. By Definition 77, and because I is a model of Axioms (4.1) and (4.2), there exists $\dot{e} \in D_I$ s.t. $I(Arg(e)) = \top$ and $I(Expl(e)) = \top$. Using Formulas (5.7), we thus have $I(ElemFixed(e)) = \top$. Then, Formula (5.3a) gives us $I(ExplEF(e)) = \top$, which, using again Formulas (5.7), results in $I(Selected(e)) = \top$ or $I(IsDefended(e)) = \top$ or $I(IsAttackerOfDefended(e)) = \top$. If $I(Selected(e)) = \top$, because I is a model of Axioms (5.1), we deduce $e \in S$, a contradiction. If $I(IsDefended(e)) = \top$, with a similar reasoning as in the previous points, we deduce that $e \in \mathscr{R}^{+2}(S)$, another contradiction. If $I(IsAttackerOfDefended(e)) = \top$, with a similar reasoning as in the previous points, we deduce that $e \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, again a contradiction. The case where $S \cup \mathscr{R}^{+2}(S) \cup \mathscr{R}^{-1}(\mathscr{R}^{+2}(S)) \not\subseteq \mathscr{A}_I$ is dealt with using the reverse reasoning.

Assume secondly that $X \not\subseteq \mathscr{R}_I$. So there exists $e \in \mathscr{R}$ s.t. $s(e) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, $t(e) \in \mathscr{R}^{+2}(S)$, and $e \notin \mathscr{R}_I$. As $e \in \mathscr{R}$ and $e \notin \mathscr{R}_I$, and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \bot$. Thus, using Formulas (5.7) we deduce $I(ElemVar(e)) = \top$. Moreover, as I is a model of Axioms (4.3), (4.11) and (4.12), there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_2)) = \top$ and $I(T(e, e_1)) = \top$. With a similar reasoning as in the previous points, from $s(e) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ and $s(e) = e_2$, we get $I(IsAttackerOfDefended(e_2)) = \top$, and from $t(e) \in \mathscr{R}^{+2}(S)$ and $t(e) = e_1$, we get $I(IsDefended(e_2)) = \top$. Then, Formulas (5.7) yield $I(NecessaryEV(e)) = \top$ and thus, by Formula (5.3b) we obtain $I(Expl(e)) = \top$, a contradiction.

Assume thirdly that $\mathscr{R}_I \not\subseteq X \cup Y$. So there exists $e \in \mathscr{R}_I$ s.t. $e \notin X$ and $e \notin Y$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.7), we have $I(ElemVar(e)) = \top$, so by Formula (5.3c), $I(NecessaryEV(e)) = \top$ or $I(AdditionalEV(e)) = \top$. If $I(NecessaryEV(e)) = \top$, by formulas (5.7) there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_2)) = \top$, $I(T(e, e_1)) = \top$, $I(IsDefended(e_1)) = \top$ and $I(IsAttackerOfDefended(e_2)) = \top$. Because I is a model of Axioms (4.3), (4.2), (4.11) and (4.12), we have $s(e) = e_2$ and $t(e) = e_1$. In addition, with a similar reasoning as in the previous points, with $I(IsDefended(e_1)) = \top$ and $I(T(e, e_1)) = \top$ we deduce that $t(e) \in \mathscr{R}^{+2}(S)$, and with $I(IsAttackerOfDefended(e_2)) = \top$ and $I(S(e, e_2)) = \top$ we deduce that $s(e) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$. As such, we have $e \in X$, a contradiction. In the case where $I(AdditionalEV(e)) = \top$, with a similar reasoning, we deduce that $s(e) \in S$ and $t(e) \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, so $e \in Y$, a contradiction as well.

Finally, assume that there exists $e \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$ with $e \in \mathscr{R}^{+1}(S)$ and s.t. there exists no $\alpha \in \mathscr{R}_I$ with $t(\alpha) = e$ and $s(\alpha) \in S$. Using Axioms (4.1), (4.2), (4.3) and (5.1), there exist $\dot{e}, \dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Arg(e)) = \top$, $I(Att(e_1)) = \top$, $I(Arg(e_2)) = \top$, $I(T(e_1, e)) = \top$,

$I(S(e_1, e_2)) = \top$, $I(Selected(e_2)) = \top$. In addition, since $e \in \mathscr{R}^{-1}(\mathscr{R}^{+2}(S))$, with a similar reasoning as in the previous points, we deduce that $I(IsAttackerOfDefended(e)) = \top$. As I is a model of formulas (5.7), we deduce that $I(ParticularEF(e)) = \top$. In addition, I is a model of the formula $\exists \alpha, a\ (Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. Since I is a model of Formula (5.3d), I is thus also a model of the formula $\exists \beta, b\ (Att(\beta) \wedge T(\beta, e) \wedge S(\beta, a) \wedge Selected(a) \wedge Expl(\beta))$. In other terms, there exist $\dot{e_1'}, \dot{e_2'} \in D_I$ s.t. $I(Att(e_1')) = \top$, $I(T(e_1', e)) = \top$, $I(S(e_1', e_2')) = \top$, $I(Selected(e_2')) = \top$ and $I(Expl(e_1')) = \top$. So, using axioms (4.1), (4.3), (4.2) and (5.1), and Definition 77 we deduce that there exists $e_1' \in \mathscr{R}_I$ s.t. $t(e_1') = e$, $s(e_1') = e_2'$ and $e_2' \in S$, a contradiction.

5. Consider an answer to Q_{CA}^{Ext} for S on $\mathcal{A}$ $(\mathscr{A}', \mathscr{R}', \varnothing, \mathscr{A}' \cup \mathscr{R}', s', t')$ where $\mathscr{A}' \subseteq \mathscr{A}$, $\mathscr{R}' \subseteq \mathscr{R}$, $s' : \mathscr{R}' \mapsto \mathscr{A}'$ and $t' : \mathscr{R}' \mapsto \mathscr{A}'$. A Herbrand interpretation I of $\Sigma_2(\mathcal{A}, S)\{(5.8), (4.11), (4.12)\}$ can be defined as follows:

   - For any $\dot{e}$ in D_I, $I(Arg(e)) = \top$ iff $e \in \mathscr{A}$, $I(Att(e)) = \top$ iff $e \in \mathscr{R}$ and $I(Sup(e)) = \bot$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(S(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = s(e_1)$
   - For any $\dot{e_1}, \dot{e_2}$ in D_I, $I(T(e_1, e_2)) = \top$ iff $e_1 \in \mathscr{R}$ and $e_2 = t(e_1)$
   - For any $\dot{e}$ in D_I, $I(PrimaFacie(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(Selected(e)) = \top$ iff $e \in S$
   - For any $\dot{e}$ in D_I, $I(Expl(e)) = \top$ iff $e \in \mathscr{A}' \cup \mathscr{R}'$
   - For any $\dot{e}$ in D_I, $I(ElemFixed(e)) = \top$ iff $e \in \mathscr{A}$
   - For any $\dot{e}$ in D_I, $I(ElemVar(e)) = \top$ iff $e \in \mathscr{R}$
   - For any $\dot{e}$ in D_I, $I(ExplEF(e)) = \top$
   - For any $\dot{e}$ in D_I, $I(NecessaryEV(e)) = \bot$
   - For any $\dot{e}$ in D_I, $I(AdditionalEV(e)) = \top$ iff there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \bot$
   - For any $\dot{e}$ in D_I, $I(ParticularEF(e)) = \top$ iff $I(Arg(e)) = \top$ and $I(Selected(e)) = \bot$

   With this definition, $\mathscr{A}_I = \mathscr{A}'$ and $\mathscr{R}_I = \mathscr{R}'$. As such, we have $s_I : \mathscr{R}' \mapsto \mathscr{A}'$ and $t_I : \mathscr{R}' \mapsto \mathscr{A}'$. Moreover, using Definition 77 and this definition of I, we have for $\alpha \in \mathscr{R}'$, $s_I(\alpha) = x$ iff $s(\alpha) = x$ and $t_I(\alpha) = y$ iff $t(\alpha) = y$. Thus, we can deduce that $s_I = s'$ and $t_I = t'$. It remains to prove that I is a model of $\Sigma_2(\mathcal{A}, S)\{(5.8), (4.11), (4.12)\}$.

   I is obviously a model of Axioms (4.1), (4.2), (4.3) and (5.1), and of formulas (5.8). In addition, by definition of I, and because $\mathcal{A}$ is an AF, I is a model of Axioms (4.11) and (4.12).

   Consider Formula (5.3a) and let $\dot{e}$ in D_I such that $I(ElemFixed(e)) = \top$. By definition of I, $e \in \mathscr{A}$. In addition, still by definition of I, $I(ExplEF(e)) = \top$. Thus, we must prove that $I(Expl(e)) = \top$. By Definition 41, $\mathscr{A}' = \mathscr{A}$, so $e \in \mathscr{A}'$. By definition of I, it follows that $I(Expl(e)) = \top$. So, I is a model of Formula (5.3a).

   Consider Formula (5.3b) and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. By definition of I, $e \in \mathscr{R}$. In addition, still by definition of I, $I(NecessaryEV(e)) = \bot$. So, I is a model of Formula (5.3b).

   Consider Formula (5.3c) and let $\dot{e}$ in D_I such that $I(ElemVar(e)) = \top$. Suppose that $I(Expl(e)) = \top$. By definition of I, $e \in \mathscr{A}' \cup \mathscr{R}'$, and since $e \in \mathscr{R}$, $e \in \mathscr{R}'$. Then, by Definition 41, $s(e) \in S$ and $t(e) \notin S$. Let $e_1 = s(e)$ and $e_2 = t(e)$. Since I is a model of Axioms (4.2) and (4.3), there exist $\dot{e_1}$ and $\dot{e_2}$ in D_I such that $I(S(e, e_1)) = \top$ and $I(T(e, e_2)) = \top$. Moreover, as I is a model of Axioms (4.11) and

(4.12), $\dot{e_1}$ and $\dot{e_2}$ are unique. As $e_1 \in S$ and $e_2 \notin S$, by definition of I, we have $I(Selected(e_1)) = \top$ and $I(Selected((e_2)) = \bot$. Thus, by definition of I, $I(AdditionalEV(e)) = \top$. So, I is a model of Formula (5.3c).

Consider Formula (5.3d), let $\dot{e}$ in D_I such that $I(ParticularEF(e)) = \top$ and suppose that there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Att(e_1)) = \top$, $I(Selected(e_2)) = \top$, $I(T(e_1, e)) = \top$ and $I(S(e_1, e_2)) = \top$. In other terms, I is a model of the formula $\exists \alpha, a(Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. By definition of I, this means that $I(Arg(e)) = \top$ and $I(Selected(e)) = \bot$, so $e \in \mathscr{A}$, $e \notin S$, $e_1 \in \mathscr{R}$ and $e_2 \in S$. Still by definition of I, we deduce that $e = t(e_1)$ and $e_2 = s(e_1)$. Thus, $s(e_1) \in S$, so $e \in \mathscr{R}^{+1}(S)$. As such, by Definition 41, there exists $e'_1 \in \mathscr{R}'$ s.t. $t(e'_1) = e$ and $s(e'_1) = e'_2$ with $e'_2 \in S$. By definition of I and because I is a model of Axioms (4.1), (4.3) and (4.2), we thus know that there exist $\dot{e'_1}, \dot{e'_2} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(Expl(e'_1)) = \top$, $I(Selected(e'_2)) = \top$, $I(T(e'_1, e)) = \top$ and $I(S(e'_1, e'_2)) = \top$. So I is a model of the formula $\exists \beta, b(Att(\beta) \wedge T(\beta, e) \wedge S(\beta, b) \wedge Selected(b) \wedge Expl(\beta))$. And so I is a model of of Formula (5.3d).

$\Leftarrow$ Let I be a Herbrand model of $\Sigma_2(\mathcal{A}, S)\{(5.8), (4.11), (4.12)\}$. Consider $X = \{\alpha \in \mathscr{R} | s(\alpha) \in S, t(\alpha) \notin S\}$ and suppose that $(\mathscr{A}_I, \mathscr{R}_I, \varnothing, \mathscr{A}_I \cup \mathscr{R}_I, s_I, t_I)$ is not an answer to Q^{Ext}_{CA} for S on $\mathcal{A}$.

Assume firstly that $\mathscr{A}_I \neq A$. Thus, either $\mathscr{A}_I \not\subseteq \mathscr{A}$ or $\mathscr{A} \not\subseteq \mathscr{A}_I$. By Definition 77, we obviously have $\mathscr{A}_I \subseteq \mathscr{A}$, so it must be the case that $\mathscr{A} \not\subseteq \mathscr{A}_I$. So there exists $e \in \mathscr{A}$ s.t. $e \notin \mathscr{A}_I$. As $e \in \mathscr{A}$ and because I is a model of Axioms (4.1) and (4.2), there exists $\dot{e} \in D_I$ with $I(Arg(e)) = \top$. Moreover, since $e \notin \mathscr{A}_I$, by Definition 77, we know that $I(Expl(e)) = \bot$. However, using Since I is a model of formulas (5.8), we have $I(ElemFixed(e)) = \top$ and $I(ExplEF(e)) = \top$. By using formula (5.3a), we deduce $I(Expl(e)) = \top$, a contradiction.

Assume secondly that $\mathscr{R}_I \not\subseteq X$. So there exists $e \in \mathscr{R}_I$ s.t. $s(e) \notin S$ or $t(e) \in S$. As $e \in \mathscr{R}_I$ and because I is a model of Axioms (4.1) and (4.2), by Definition 77, there exists $\dot{e} \in D_I$ with $I(Att(e)) = \top$ and $I(Expl(e)) = \top$. Since I is a model of formulas (5.8), we have $I(ElemVar(e)) = \top$ and $I(NecessaryEV(e)) = \bot$, and so by Formula (5.3c), $I(AdditionalEV(e)) = \top$. By using again formulas (5.8), we deduce that there exist $\dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(S(e, e_1)) = \top$, $I(T(e, e_2)) = \top$, $I(Selected(e_1)) = \top$ and $I(Selected(e_2)) = \bot$. In addition, I is a model of Axioms (4.3), (4.11) and (4.12), so $e_1 = s(e)$ and $e_2 = t(e)$. Finally, as I is a model of Axiom (5.1), we have $e_1 \in S$ and $e_2 \notin S$, so $s(e) \in S$ and $t(e) \notin S$, a contradiction.

Finally, assume that there exists $e \in \mathscr{A} \setminus S$ with $e \in \mathscr{R}^{+1}(S)$ and s.t. there exists no $\alpha \in \mathscr{R}_I$ with $t(\alpha) = e$ and $s(\alpha) \in S$. Using Axioms (4.1), (4.2), (4.3) and (5.1), there exist $\dot{e}, \dot{e_1}, \dot{e_2} \in D_I$ s.t. $I(Arg(e)) = \top$, $I(Selected(e)) = \bot$, $I(Att(e_1)) = \top$, $I(Arg(e_2)) = \top$, $I(T(e_1, e)) = \top$, $I(S(e_1, e_2)) = \top$, $I(Selected(e_2)) = \top$. As I is a model of formulas (5.8), we deduce that $I(ParticularEF(e)) = \top$. In addition, I is a model of the formula $\exists \alpha, a\ (Att(\alpha) \wedge T(\alpha, e) \wedge S(\alpha, a) \wedge Selected(a))$. Since I is a model of Formula (5.3d), I is thus also a model of the formula $\exists \beta, b\ (Att(\beta) \wedge T(\beta, e) \wedge S(\beta, a) \wedge Selected(a) \wedge Expl(\beta))$. In other terms, there exist $\dot{e'_1}, \dot{e'_2} \in D_I$ s.t. $I(Att(e'_1)) = \top$, $I(T(e'_1, e)) = \top$, $I(S(e'_1, e'_2)) = \top$, $I(Selected(e'_2)) = \top$ and $I(Expl(e'_1)) = \top$. So, using Axioms (4.1), (4.3), (4.2) and (5.1), and Definition 77 we deduce that there exists $e'_1 \mathscr{R}_I$ s.t. $t(e'_1) = e$, $s(e'_1) = e'_2$ and $e'_2 \in S$, a contradiction.

□